# ZXR10 8900 Series
## 10 Gigabit Routing Switch
## User Manual (Basic Configuration Volume)

Version 2.8.02.C

ZTE CORPORATION
ZTE Plaza, Keji Road South,
Hi-Tech Industrial Park,
Nanshan District, Shenzhen,
P. R. China
518057
Tel: (86) 755 26771900
Fax: (86) 755 26770801
URL: http://ensupport.zte.com.cn
E-mail: support@zte.com.cn

LEGAL INFORMATION

Copyright © 2006 ZTE CORPORATION.

The contents of this document are protected by copyright laws and international treaties. Any reproduction or distribution of this document or any portion of this document, in any form by any means, without the prior written consent of ZTE CORPORATION is prohibited. Additionally, the contents of this document are protected by contractual confidentiality obligations.

All company, brand and product names are trade or service marks, or registered trade or service marks, of ZTE CORPORATION or of their respective owners.

This document is provided "as is", and all express, implied, or statutory warranties, representations or conditions are disclaimed, including without limitation any implied warranty of merchantability, fitness for a particular purpose, title or non-infringement. ZTE CORPORATION and its licensors shall not be liable for damages resulting from the use of or reliance on the information contained herein.

ZTE CORPORATION or its licensors may have current or pending intellectual property rights or applications covering the subject matter of this document. Except as expressly provided in any written license between ZTE CORPORATION and its licensee, the user of this document shall not acquire any license to the subject matter herein.

ZTE CORPORATION reserves the right to upgrade or make technical change to this product without further notice.

Users may visit ZTE technical support website http://ensupport.zte.com.cn to inquire related information.

The ultimate right to interpret this product resides in ZTE CORPORATION.

**Revision History**

| Revision No. | Revision Date | Revision Reason |
|---|---|---|
| R1.0 | July. 31, 2009 | First Release |

Serial Number: sjzl20093837

# Contents

**About This Manual** ..... i
**Safety Instructions** ..... 1
Safety Introduction ..... 1
Safety Description ..... 1
**Usage and Operation** ..... 3
Configuration Modes ..... 3
Configuring Serial Interface Connection ..... 4
Configuring Telnet Connection ..... 6
Configuring SSH Connection ..... 9
Configuring SNMP Connection ..... 11
Command Modes ..... 12
Command Line Usage ..... 14
Online Help ..... 14
Command Abbreviation ..... 15
Command History ..... 15
**System Management** ..... 17
File System Management ..... 17
File System Overview ..... 17
Operating File System Management ..... 18
FTP/TFTP Connection Configuration ..... 19
Configuring a Switch as FTP Client Terminal ..... 20
Configuring a Switch as TFTP Client Terminal ..... 21
File Backup and Restoration ..... 23
Backing up Configuration File ..... 23
Restoring Configuration File ..... 23
Backing up System Software Version ..... 23
Restoring System Software Version ..... 24
Ststem Software Version Upgrade ..... 24
Upgrading Version at Abnormality ..... 24
Upgrading Version at Normality ..... 26
Upgrading Version without Interrupting System ..... 27

System Parameter Configuration....28
Configuring a Hostname....28
Configuring a Welcome Message....29
Configuring a Password of Privileged Mode....29
Configuring Telnet Username and Password....29
Configuring System Time....30
Configuring Version Load Selection....30
Saving Command Log File....31
Configuring Saving Time of Alarm Log....32
System Information View....33
Viewing Hardware and Software Versions....33
Viewing Current Running Configuration Information....33
Viewing CPU Information....34
Viewing Boot Information of Current Running Board....34
Viewing System Diagnosis Information....34
**CLI Privilege Classification....37**
CLI Privilege Classification Overview....37
Configuring CLI Privilege Classification....38
Configuring Telnet User....38
Configuring an Enabling Password....39
Configuring Privilege Level of a Command....40
CLI Privilege Classification Configuration Example....42
Maintenance and Diagnosis of CLI Privilege Classification....42
**Port Configuration....43**
Port Basic Configuration....43
Port Basic Configuration Overview....43
Enabling an Ethernet Port....44
Enabling Auto-Negotiation....44
Configuring Duplex Mode....45
Configuring Ethernet Port Rate....45
Configuring Traffic Control....46
Allowing Jumbo-Frame....46
Configuring Broadcast Storm Suppression....47
Configuring Multicast Suppression....47
Configuring Unknown Unicast Suppression....48
Enabling Fast Port Detection Function....48
Configuring FEFI Function....49

Configuring TCP Rate Limit....49
Configuring Switch of Optical or Electrical Port....49
Viewing Port Information....49
Diagnosing and Testing Link....51
Port Mirroring Configuration....52
Port Mirroring Overview....52
Configuring Port Mirroring....52
Port Mirroring Configuration Example....52
ERSPAN Configuration....54
ERSPAN Overview....54
Configuring ERSPAN....55
Establishing One ERSPAN Session....55
Adding Source or Destination Port to Session Entry....55
Displaying Session Details Configured by User....55
ERSPAN Configuration Example....55
Port Loop Detection Configuration....56
Port Loop Detection Overview....56
Configuring Port Loop Detection....56
Port Loop Detection Configuration Example....57
**Network Protocol Configuration....59**
IP Address Configuration....59
IP Address Overview....59
Configuring IP Address....61
IP Address Configuration Example....61
ARP Configuration....61
ARP Overview....61
Configuring ARP....62
ARP Configuration Example....62
ARP Query Example....63
**DHCP Configuration....65**
DHCP Overview....65
DHCP Snooping Overview....66
Configuring DHCP....66
Configuring DHCP Server....66
Configuring DHCP Relay....67
Configuring DHCP Snooping....67
DHCP Configuration Examples....68
DHCP Server Configuration Example....68
DHCP Relay Configuration Example....69

DHCP Snooping Preventing False DHCP Server Configuration Example ..... 70
DHCP Snooping Preventing Static IP Configuration Example ..... 70
DHCP Maintenance and Diagnosis ..... 71
**VRRP Configuration ..... 73**
VRRP Overview ..... 73
Configuring VRRP ..... 74
VRRP Configuration Examples ..... 74
Basic VRRP Configuration Example ..... 74
Symmetric VRRP Configuration Example ..... 75
VRRP Maintenance and Diagnosis ..... 76
**ACL Configuration ..... 77**
ACL Overview ..... 77
NP-Based ACL Overview ..... 78
Configuring ACLs ..... 79
Defining ACLs ..... 79
Defining Standard ACL ..... 79
Defining Extended ACL ..... 80
Defining Layer 2 ACL ..... 81
Defining Hybrid ACL ..... 81
Defining Standard IPv6 ACL ..... 82
Defining Extended IPv6 ACL ..... 82
Defining Customized ACL ..... 83
Configuring Time Range ..... 83
Applying ACL to Physical Port ..... 84
Applying ACL to Virtual Port ..... 85
Configuring Event Linkage ACL Rule ..... 85
Applying NP-Based ACL ..... 87
ACL Configuration Example ..... 88
ACL Maintenance and Diagnosis ..... 89
**QoS Configuration ..... 91**
QoS Overview ..... 91
Traffic Classification ..... 92
Traffic Monitoring ..... 92
Traffic Shaping ..... 93
Queue Scheduling and Default 802.1p ..... 93
Policy Routing ..... 94
Priority Mark ..... 94
Traffic Mirroring ..... 95

Traffic Statistics ..... 95
Queue-Based Bandwidth Upper and Lower Threshold ..... 95
HQoS ..... 95
Configuring QoS ..... 96
Configuring Traffic Monitoring ..... 96
Configuring Traffic Rate Limit ..... 97
Configuring Layer 3 Rate Limit ..... 97
Configuring Queue Scheduling ..... 98
Configuring Policy Routing ..... 99
Configuring Priority Mark ..... 99
Configuring Tail Discarding ..... 100
Configuring COS Discarding Priority Mapping ..... 100
Configuring COS Local Priority Mapping ..... 101
Configuring DSCP Priority Mapping ..... 101
Configuring Traffic Mirroring ..... 102
Configuring Traffic Statistics ..... 102
Configuring Queue-Based Bandwidth Upper and Lower Threshold ..... 103
Configuring HQoS ..... 103
Configuring Traffic Class ..... 103
Configuring WRED Policy ..... 104
Configuring WFQ Policy ..... 105
Configuring Traffic Shaping ..... 105
Configuring HQoS Policy ..... 106
QoS Configuration Examples ..... 109
Typical QoS Configuration Example ..... 109
Policy Routing Configuration Example ..... 111
QoS Maintenance and Diagnosis ..... 111
**DOT1x Configuration ..... 113**
DOT1x Overview ..... 113
Configuring DOT1x ..... 114
Configuring AAA ..... 114
Configuring DOT1x Parameters ..... 115
Configuring Local Authentication User ..... 115
Managing DOT1x Authentication User ..... 116
DOT1x Configuration Examples ..... 117
Dot1x Radius Authentication Application ..... 117
Dot1x Relay Authentication Application ..... 118
Dot1x Local Authentication Application ..... 119

DOT1x Maintenance and Diagnosis ..... 120
**Cluster Management Configuration ..... 121**
Cluster Management Overview ..... 121
Configuring Cluster Management ..... 123
Enabling ZDP ..... 123
Enabling ZTP ..... 124
Setting up a Cluster ..... 124
Maintaining a Cluster ..... 125
Configuring Cluster Operation Commands ..... 125
Cluster Management Configuration Example ..... 126
Cluster Management Maintenance and Diagnosis ..... 126
**Network Management Configuration ..... 129**
NTP Configuration ..... 129
NTP Overview ..... 129
Configuring NTP ..... 129
NTP Configuration Example ..... 130
RADIUS Configuration ..... 130
Radius Overview ..... 130
Configuring a RADIUS Accounting Group ..... 130
Configuring a RADIUS Authentication Group ..... 131
Configuring RADIUS Parameters ..... 131
Viewing RADIUS Information ..... 132
RADIUS Configuration Example ..... 132
SNMP Configuration ..... 133
SNMP Overview ..... 133
Configuring SNMP ..... 133
SNMP Configuration Example ..... 134
RMON Configuration ..... 134
RMON Overview ..... 134
Configuring RMON ..... 135
RMON Configuration Example ..... 135
SysLog Configuration ..... 136
SysLog Overview ..... 136
Configuring SysLog ..... 137
SysLog Configuration Example ..... 137
LLDP Configuration ..... 138
LLDP Overview ..... 138
Configuring LLDP ..... 139
LLDP Configuration Example ..... 139
**IPTV Configuration ..... 141**

IPTV Overview ..... 141
Configuring IPTV ..... 141
Configuring IPTV Global Parameters ..... 141
Configuring Global Parameters of IPTV Preview ..... 142
Configuring IPTV CDR Parameters ..... 142
Configuring IPTV Channels ..... 143
Configuring IPTV Service Package ..... 143
Configuring IPTV Preview Template ..... 144
Configuring CAC ..... 144
Configuring IPTV Fast Leave ..... 145
Managing IPTV Users ..... 145
IPTV Configuration Example ..... 145
IPTV Maintenance and Diagnosis ..... 146
**VBAS Configuration ..... 149**
VBAS Overview ..... 149
Configuring VBAS ..... 149
VBAS Configuration Example ..... 150
VBAS Maintenance and Diagnosis ..... 150
**CPU Attack Protection Configuration ..... 151**
CPU Attack Protection Overview ..... 151
CPU Attack Protection Principle ..... 152
Configuring CPU Attack Protection ..... 152
Configuring IPv4 Protocol Protection ..... 152
Configuring IPv6 Protocol Protection ..... 153
Configuring Layer 2 Protocol Protection ..... 154
CPU Attack Protection Configuration Examples ..... 154
**URPF Configuration ..... 157**
URPF Overview ..... 157
Configuring URPF ..... 158
URPF Configuration Example ..... 159
URPF Maintenance and Diagnosis ..... 160
**IPFIX Configuration ..... 161**
IPFIX Overview ..... 161
IPFIX Overview ..... 161
Sampling ..... 162
Timeout Management ..... 162
Data Output ..... 163
Configuring IPFIX ..... 163
Basic Configuration ..... 163

Enabling/Disabling IPFIX Module ..... 163
Setting IPFIX Memory Entries ..... 163
Setting Aging Time of Active Stream ..... 163
Setting Aging Time of Inactive Stream ..... 164
Setting Sampling Rate ..... 164
Setting NM Server Address and L4 Port ID ..... 164
Setting Source Address for Network Device Sending Packets ..... 164
Setting Template Refresh Rate ..... 164
Configuring TOPN ..... 165
Template Configuration ..... 165
Setting Template ..... 165
Setting Data Field Contained in Template Packet ..... 165
Deleting Template ..... 165
Running Template ..... 165
IPFIX Configuration Example ..... 166
IPFIX Maintenance and Diagnosis ..... 166
**Figures ..... 169**
**Tables ..... 171**
**List of Glossary ..... 173**

# About This Manual

**Purpose**

This manual provides procedures and guidelines that support the operation of ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch.

**Intended Audience**

This manual is intended for engineers and technicians who perform operation activities on ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch.

**What Is in This Manual**

This manual contains the following chapters:

**TABLE 1 CHAPTER SUMMARY**

| Chapter | Summary |
|---|---|
| Chapter 1 Safety Instructions | This chapter describes the safety instructions and signs |
| Chapter 2 Usage and Operation | This chapter describes ZXR10 8912/8908/8905/8902 configuration mode in common use |
| Chapter 3 System Management | This chapter introduces file system management, file backup and restoration, software version upgrade |
| Chapter 4 CLI Privilege Classification | This chapter describes CLI privilege classification and configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 5 Port Configuration | This chapter describes the configuration of ZXR10 8912/8908/8905/8902 port parameters and port mirroring function |
| Chapter 6 Network Protocol Configuration | This chapter describes IP address configuration and ARP configuration |
| Chapter 7 DHCP Configuration | This chapter introduces DHCP and related configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 8 VRRP Configuration | This chapter describes Virtual Router Redundancy Protocol (VRRP) on ZXR10 8912/8908/8905/8902 |
| Chapter 9 ACL Configuration | This chapter introduces ACL and related configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 10 QoS Configuration | This chapter introduces QoS and related configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 11 DOT1x Authentication Configuration | This chapter introduces DOT1x Authentication configuration on ZXR10 8912/8908/8905/8902 |

| Chapter | Summary |
|---|---|
| Chapter 12 Cluster Management Configuration | This chapter introduces cluster management configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 13 Network Management Configuration | This chapter introduces Network management configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 14 IPTV Configuration | This chapter describes IPTV configuration, maintenance and diagnosis for ZXR10 8912/8908/8905/8902 |
| Chapter 15 VBAS Configuration | This chapter describes VBAS on ZXR10 8912/8908/8905/8902 |
| Chapter 16 CPU Attack Protection Configuration | This chapter describes configuration for CPU attack protection on ZXR10 8912/8908/8905/8902 |
| Chapter 17 URPF Configuration | This chapter introduces URPF (Unicast Reverse Path Forwarding) and related configuration on ZXR10 8912/8908/8905/8902 |
| Chapter 18 UDLD Configuration | This chapter describes UDLD and configuration on ZXR10 8912/8908/8905/8902 |

**Related Documentation**

The following documentation is related to this manual:

- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch Hardware Installation Manual
- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch Hardware Manual
- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch User Manual (Basic Configuration Volume)
- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch User Manual (Ethernet Switching Volume)
- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch User Manual (IPv4 Routing Volume)
- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch User Manual (MPLS Volume)
- ZXR10 8900 Series (V2.8.02.C) 10 Gigabit Routing Switch User Manual (IPv6 Volume)

Chapter 1

# Safety Instructions

**Table of Contents**

Safety Introduction ..... 1
Safety Description ..... 1

## Safety Introduction

In order to operate the equipment in a proper way, follow these instructions:

- Only qualified professionals are allowed to perform installation, operation and maintenance due to the high temperature and high voltage of the equipment.
- Observe the local safety codes and relevant operation procedures during equipment installation, operation and maintenance to prevent personal injury or equipment damage. Safety precautions introduced in this manual are supplementary to the local safety codes.
- ZTE bears no responsibility in case of universal safety operation requirements violation and safety standards violation in designing, manufacturing and equipment usage.

## Safety Description

Contents deserving special attention during configuration of ZXR10 8900 series switch are explained in the following table.

| Convention | Meaning |
|---|---|
| **Note** | Provides additional information |
| **Important** | Provides great significance or consequence |
| **Result** | Provides consequence of actions |
| **Example** | Provides instance illustration |

This page is intentionally blank.

Chapter 2

# Usage and Operation

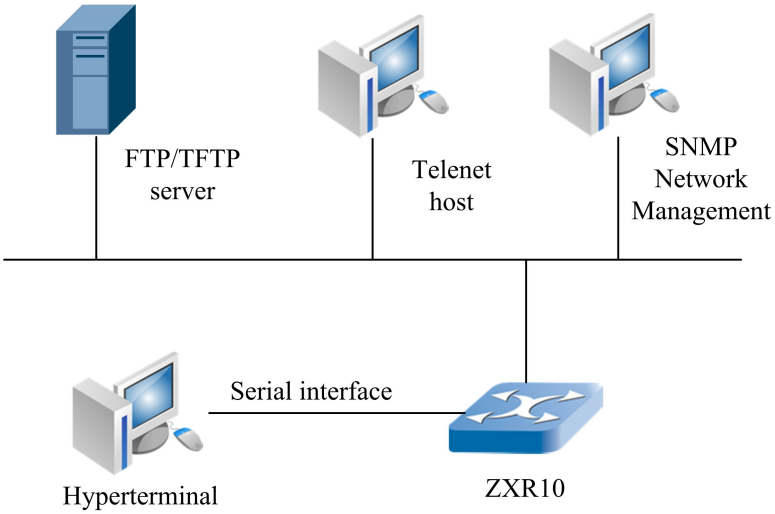
**Table of Contents**

Configuration Modes ..... 3
Command Modes..... 12
Command Line Usage ..... 14

# Configuration Modes

ZXR10 8900 series switch provides multiple configuration modes, as shown in Figure 1. User can select appropriate configuration mode according to the connected network.

**FIGURE 1 CONFIGURATION MODES**




- Serial interface connection configuration
- TELNET connection configuration
- SSH connection configuration
- FTP/TFTP connection configuration
- SNMP connection configuration

# Configuring Serial Interface Connection

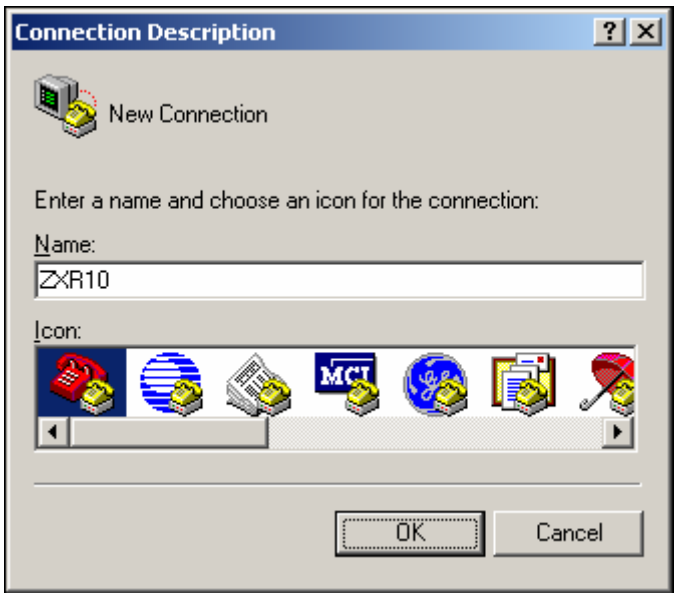
Serial interface connection configuration is the principle configuration mode of ZXR10 series switch.

Serial configuration cable is delivered with ZXR10 8900 series switch. One end is DB9 serial interface (connecting to computer serial interface). The other end is RJ45 interface (connecting to Console interface in MP board of ZXR10 8900 series switch). Serial connection configuration adopts VT100 terminal mode, using the HyperTerminal tool provided by Windows OS.

To configure serial interface connection, perform the following steps.

1. Connect the computer serial port to Console port of ZXR10 8900 series switch with serial configuration cable.
2. Open the HyperTerminal, as shown in Figure 2. Input the connection name, such as ZXR10, and select the desired icon.

**FIGURE 2 HYPERTERMINAL CONFIGURATION 1**




3. Click **Ok**. A window appears, as shown in Figure 3. Select COM1 as COM port in the **Connect using** field.

**FIGURE 3 HYPERTERMINAL CONFIGURATION 2**

4. Click **Ok**. **COM port attribute setup** window appears, as shown in Figure 4. Fill in the parameter values, as shown in Table 3.

**FIGURE 4 HYPERTERMINAL CONFIGURATION 3**

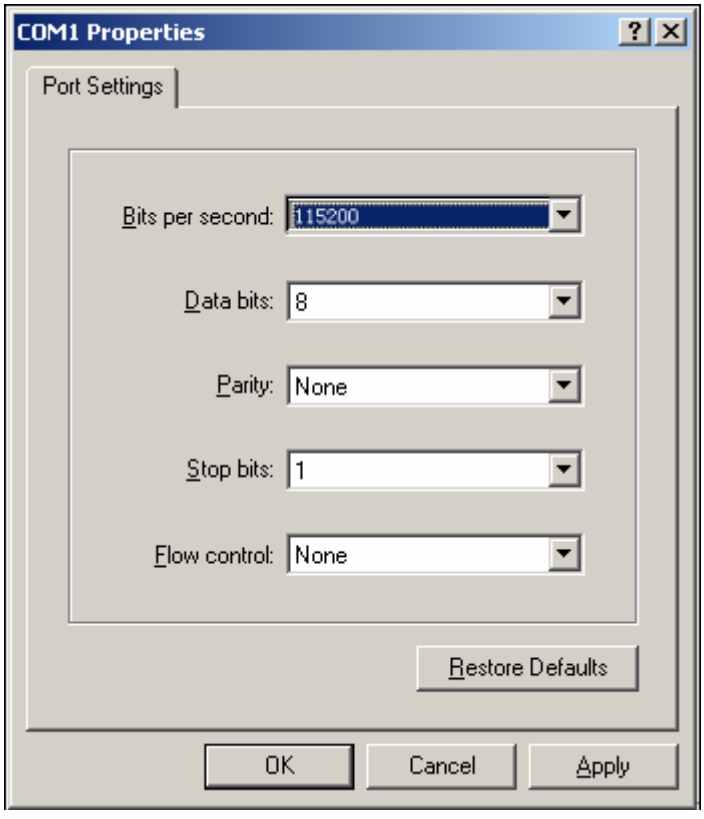


**TABLE 3 PARAMETER VALUES**

| Parameters | Values |
|---|---|
| Bits per second | 115200 |
| Data bit | 8 |
| Parity | None |
| Stop bit | 1 |
| Flow control | None |

**Note:**

If the switch fails to be connected, set the value of bits per second to 9600.

5. Click **Ok** to complete setting. ZXR10 8900 series switch configuration window appears. At this point start command operation.

**Result:** Serial interface connection has been configured.

# Configuring Telnet Connection

ZXR10 8900 series switch can be configured by Telnet locally or remotely. Telnet configuration is the principal mode that is used to configure ZXR10 8900 series switch remotely.

Username and password must be set in the switch to prevent illegal users from accessing the switch by Telnet. Only the users with valid username and password could login to the device. Use the following command to configure username and password.

| Command | Function |
|---|---|
| `ZXR10(config)#`**username** *<username>* **password** *<password>* | This configures username and password of Telnet login |

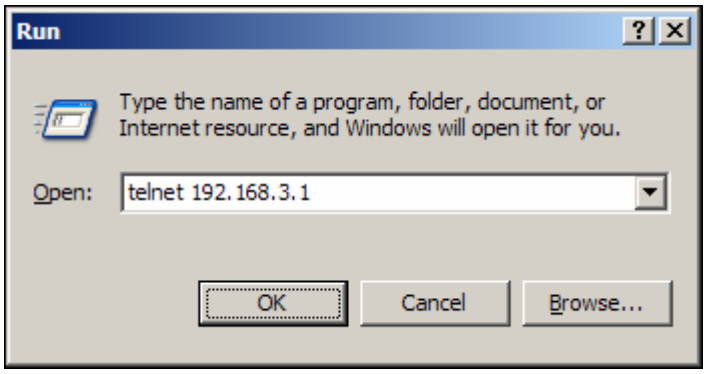
**Configuring Telnet Connection through Management Port**

To configure telnet connection through management Ethernet interface (10/100Base-TX) on main board, perform the following steps:

1. Configure IP address of management port through Console port.
2. Configure username and password of Telnet login through Console port.
3. Use straight-through Ethernet cable to connect host network interface and switch management Ethernet interface.
4. Set the IP address of the host that is a part of the same network segment with the switch management Ethernet interface.

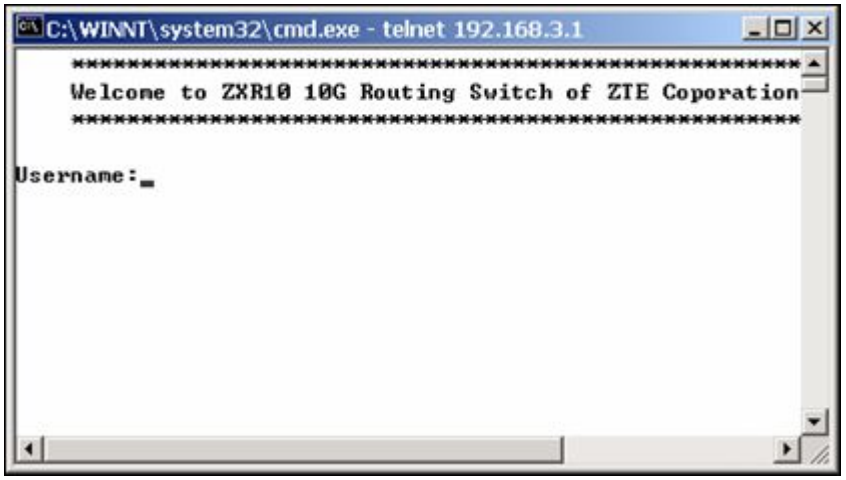
5. Execute **telnet** command in the host. Input the IP address of switch management Ethernet port, as shown in Figure 5.

**FIGURE 5 RUNNING TELNET**




6. Click **OK**. A window appears, as shown in Figure 6.

**FIGURE 6 TELNET LOGIN SCHEMATIC DIAGRAM**




7. Input valid username and password to enter switch configuration mode.

**Note:**

- ZXR10 8900 series switch allows up to four Telnet users logging in simultaneously. If "**" appears after inputting username and password, it indicates that the number of users reaches the limit, please retry later or re-login after logging out other users.
- When users perform Telnet configuration through management port connecting to the switch, the IP address of management port cannot be modified or deleted, otherwise, Telnet will be disconnected.

**Configuring Telnet Connection through Host**

To configure a telnet connection to a switch through a VLAN port, perform the following steps.

1. Configure IP addresses of VLAN and VLAN interface through Console port.
2. Configure username and password of Telnet login through Console port.
3. Connect the host network interface to the Ethernet port of switch.
4. Set IP address of host, enabling the host to ping the IP address of VLAN interface in the switch successfully.
5. Execute **telnet** command in the host. Input the IP address of VLAN interface, login to the switch. For the detailed procedures, please refer to Configuring Telnet Connection through Management Port.

**Configuring Telnet Connection through Other Devices (Such as Switch or Router)**

To configure telnet connection through other devices (such as switch and router), perform the following steps.

1. Configure IP address of VLAN and VLAN interface through Console port.
2. Configure username and password of Telnet login through Console port.
3. Take a router connected to a switch as an example, from which, the IP address of VLAN interface can be pinged successfully.
4. Run **telnet** command in the router. Input the IP address of VLAN interface, login to the switch. For the detailed procedures, please refer to Configuring Telnet Connection through Management Port.

**Note:**

When users perform Telnet configuration through VLAN interface connecting to the switch, the IP address of VLAN and VLAN interface cannot be modified or deleted, otherwise, Telnet is disconnected.

**Configuring Limit to Telnet Connections**

The number of Telnet connections can be limited by the following command configuration to enhance system security and practicability.

| Command | Function |
|---|---|
| `ZXR10(config)#` **Line telnet** < *max-link*> | This adds limit to the number (1–16) of connected users. |

**Example**

As shown in Figure 7, one PC is connected to interface gei_1/1. To telnet switch, conduct the following configuration:

**FIGURE 7 TELNET CONNECTION LIMIT CONFIGURATION EXAMPLE**

Configuration of Switch:

```
ZXR10(config)#line telnet max-link 2
```

# Configuring SSH Connection

Telnet and FTP connections are not safe because they use the plain text to transmit the password and data on the network. This results in data to be easily intercepted by hackers. A disadvantage of the Telnet/FTP security authentication is that it is easily attacked by the man-in-the-middle. This imitates the server to receive the data transmitted by the client terminal and then imitates the client terminal to transmit data to the real server.

SSH (Secure Shell) can solve the problem. SSH establishes a secure channel for remote login and other network services in the insecure network. It encrypts and compresses the transmitted data that prevents people from getting secret information.

Two incompatible versions of SSH protocols are available:

- SSH v1.x
- SSH v2.x

ZXR10 8900 series switch supports SSH v2.0. It provides secure remote login function.

SSH falls into two parts including server and client terminal. ZXR10 8900 series switch serves as the server of SSH. Host logs in to the switch by running SSH client terminal.

To configure SSH connection, perform the following steps.

1. Use the following commands to enable SSH server function of ZXR10 8900 series switch.

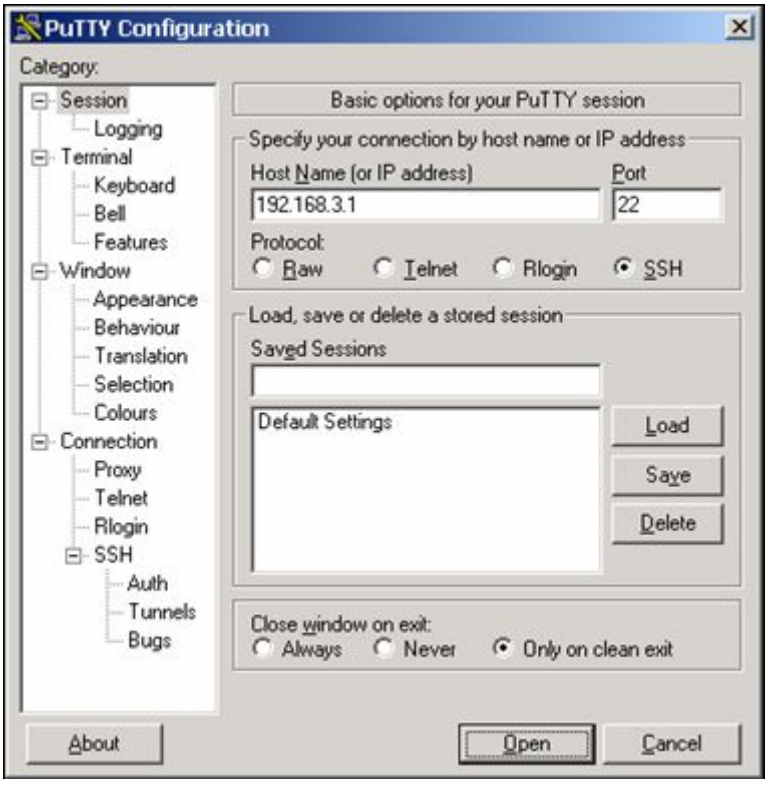
| **Command** | **Function** |
|---|---|
| `ZXR10(config)#`**ssh server enable** | This enables SSH server function |

**Note:**

The SSH server function is disabled by default.

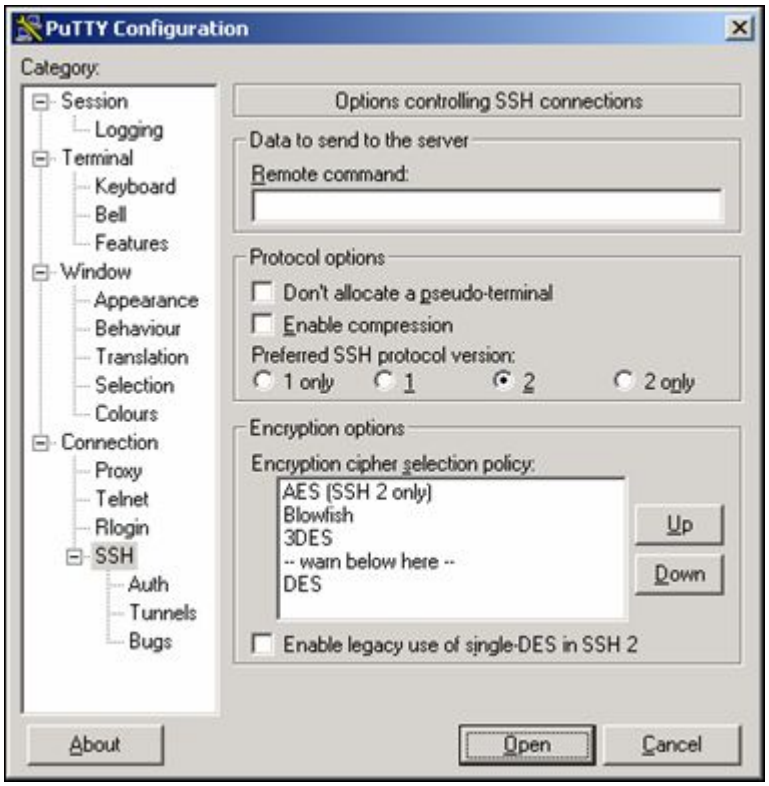
2. Connect the host network interface to the Ethernet port of the switch. Enable the host to ping the IP address of VLAN interface in the switch.
3. Run SSH client terminal software in the host
   i. Set the IP address and port number of SSH server, as shown in Figure 8.

**FIGURE 8 SETTING IP ADDRESS AND PORT OF SSH SERVER**




   ii. Set SSH version, as shown in Figure 9.

**FIGURE 9 SETTING SSH VERSION**




4. Click **Open** to login to the switch and input valid username and password.

**Result:** SSH connection has been configured.

# Configuring SNMP Connection

Simple Network Management Protocol (SNMP) is an NM protocol. With SNMP, one NM server can manage all devices in the network.

SNMP adopts management, based on server and client terminal. Background NM server serves as the SNMP server, and the foreground network equipment. ZXR10 8900 series switch serves as SNMP client terminal. Foreground and background share the same MIB management database, performing communication by SNMP protocol.

Background NM server needs installation of NM software that supports SNMP protocol. It performs management configuration over ZXR10 8900 series switch by NM software.

# Command Modes

ZXR10 8900 series switch assigns commands to different modes according to function and authority to facilitate switch configuration and management. One command can only be executed under specific mode. Input a question mark (?) under any command mode to query the applicable commands under the mode. Major command modes of ZXR10 8900 series switch are described in Table 4.

**TABLE 4 COMMAND MODES**

| Mode | Prompt | Accessing Command |
|---|---|---|
| User EXEC | ZXR10> | Access this mode directly after login |
| Privileged EXEC | ZXR10# | **enable** (User EXEC mode) |
| Global configuration | ZXR10(config)# | **configure terminal** (Privileged EXEC mode) |
| Port configuration | ZXR10(config-if)# | **interface** {<*interface-name*>\|**byname** <*by-name*>} (Global configuration mode) |
| VLAN database configuration | ZXR10(vlan)# | **vlan database** (Privileged EXEC mode) |
| VLAN configuration | ZXR10(config-vlan)# | **vlan** {<*vlan-id*>\|<*vlan-name*>} (Global configuration mode) |
| VLAN interface configuration | ZXR10(config-if)# | **interface** {**vlan** <*vlan-id*>\|<*vlan-if*>} (Global configuration mode) |
| MSTP configuration | ZXR10(config-mstp)# | **spanning-tree mst configuration** (Global configuration mode) |
| Basic ACL configuration | ZXR10(config-std-acl)# | **acl standard** {**number** <*acl-number*>\| **name** <*acl-name*>} (Global configuration mode) |
| Extended ACL configuration | ZXR10(config-ext-acl)# | **acl extend** {**number** <*acl-number*>\| **name** <*acl-name*>} (Global configuration mode) |
| L2 ACL configuration | ZXR10(config-link-acl)# | **acl link** {**number** <*acl-number*>\| **name** <*acl-name*>} (Global configuration mode) |
| Hybrid ACL configuration | ZXR10(config-hybd-acl)# | **acl hybrid** {**number** <*acl-number*>\| **name** <*acl-name*>} (Global configuration mode) |

| Mode | Prompt | Accessing Command |
|---|---|---|
| Customized ACL configuration | ZXR10(config-user-defined-acl)# | **acl user-defined** { **numberr** < *acl-number*>| **naame** < *acl-name*>| **aalliiaass**< *ACL alias*>}(Global configuration mode) |
| VRF configuration mode | ZXR10(config-vrf)# | **ip vrf** <*vrf-name*> (Global configuration mode) |
| RIP route configuration | ZXR10(config-router)# | **router rip** (Global configuration mode) |
| RIP address family configuration | ZXR10(config-router-af)# | **address-family ipv4 vrf** <*vrf-name*> (Route RIP configuration mode) |
| OSPF route configuration | ZXR10(config-router)# | **router ospf** <*process-id*>[**vrf** <*vrf-name*>] (Global configuration mode) |
| IS-IS route configuration | ZXR10(config-router)# | **router isis** [**vrf** <*vrf-name*>] (Global configuration mode) |
| BGP route configuration | ZXR10(config-router)# | **router bgp** <*as-number*> (Global configuration mode) |
| BGP address family configuration | ZXR10(config-router-af)# | **address-family vpnv4** (Route BGP configuration mode)<br>**address-family ipv4 vrf** <*vrf-name*> (BGP route configuration mode) |
| PIM-SM route configuration | ZXR10(config-router)# | **router pimsm** (Global configuration mode) |
| Route map configuration | ZXR10(config-route-map)# | **route-map** <*map-tag*>[**permit**\|**deny**][<*sequence-number*>] (Global configuration mode) |
| Diagnosis test | ZXR10(diag)# | **diagnose** (Privileged EXEC mode) |

The following commands are used to exit from different command modes:

- In privileged EXEC mode, use **disable** command to return to user EXEC mode.
- In user EXEC mode and privileged EXEC mode, use **exit** command to quit the switch; in other modes, use **exit** command to return to the previous mode.
- In the modes other than user EXEC mode and privileged EXEC mode, use **end** command or press **Ctrl+z** to return to the privileged EXEC mode.

# Command Line Usage

## Online Help

In command mode, available command list is displayed if a question mark (?) is entered that follows the system prompt. Command key word list and parameters can be obtained through online help.

- Input a question mark (?) in any command mode prompt, all commands and brief command descriptions of the mode are displayed. For example:

```
ZXR10>?
Exec commands:
    enable  Turn on privileged commands
    exit    Exit from the EXEC
    login   Login as a particular user
    logout  Exit from the EXEC
    ping    Send echo messages
    quit    Quit from the EXEC
    show    Show running system information
    telnet  Open a telnet connection
    trace   Trace route to destination
    who     List users who is logining on
ZXR10>
```

- Input a question mark (?) following character or character string, the list of commands or key words with the character or character string as the prefix are displayed. For example:

```
ZXR10#co?
configure copy
ZXR10#co
```

**Note:**

There is no space between character (Character string) and the question mark (?).

- Press **Tab** after the character, if the command or key word with the character string as the prefix is unique, align it and add a space after it. For example:

```
ZXR10#con<Tab>
ZXR10#configure
```

**Note:**

There is no space between character string and **Tab**.

- Input a question mark (?) after commands, key words and parameters. It is possible to list the key words or parameters to be input. For example:

```
ZXR10#configure ?
terminal  Enter configuration mode
ZXR10#configure
```

**Note:**

A space should be input before the question mark (?).

- If incorrect command, key words or parameters are entered, subscriber interface will provide error isolation with "^" after carriage return. "^" will appear below the first character of the input incorrect command, key word or parameter. For example:

```
ZXR10#von ter
      ^
% Invalid input detected at ' ^'  marker.
ZXR10#
```

Make use of the online help to set system clock.

```
ZXR10#cl?
clear  clock
ZXR10#clock ?
set  Set the time and date
ZXR10#clock set ?
hh:mm:ss  Current Time
ZXR10#clock set 13:32:00
% Incomplete command.
ZXR10#
```

At the end of the above example, system prompts that command is incomplete. This indicates requirement of other key words or parameters.

**Note:**

All commands in the command line operation are case-insensitive.

# Command Abbreviation

ZXR10 8900 series switch allows abbreviating commands and key word to character or character string identifying the command or key word uniquely. For example, abbreviate **show** command to **sh** or **sho**.

# Command History

User interface provides a record of up to 10 previously entered commands. This feature is particularly useful to recall long or complex commands.

Re-invoke commands from the record buffer. Execute one of the following operations.

| Operation | Description |
|---|---|
| Press **Ctrl+P** or **-** | This recalls commands in the history buffer in a forward sequence |
| Press **Ctrl+N** or ¯ | This recalls commands in the history buffer in a backward sequence |

In the privileged mode, use **show history** command to list the recently used commands.

Chapter 3

# System Management

**Table of Contents**

File System Management....17
FTP/TFTP Connection Configuration....19
File Backup and Restoration....23
Ststem Software Version Upgrade....24
System Parameter Configuration....28
System Information View....33

# File System Management

## File System Overview

On ZXR10 8900 series switch, FLASH in MP board is used as major storage device that is for storing ZXR10 8900 series switch version files and configuration files. When upgrading software version and saving configuration, an operation over FLASH is necessary.

There are three directories in Flash by default.

- IMG
- CFG
- DATA

**IMG** System mapping files (that is, image files) are stored under this directory. The extended name of the image files is .zar. The image files are dedicated compression files. Version upgrade means to change the corresponding image files under the directory.

**Note:**

Default name of ZXR10 8900 series switch software version file is zxr10.zar. If it uses other names, boot Path must be modified in boot status. Otherwise, version cannot be loaded when users start the system. It is recommended using default file name.

**CFG** This directory is for saving configuration files, whose name is startrun.dat. Information is saved in the Memory when users use command to modify the switch configuration. To prevent the configuration information loss when the device restarts, use **write**

command to write the information in the Memory into FLASH, and save the information in the startrun.dat file. If it is necessary to clear the old configuration in the switch to reconfigure data, use **delete** command to delete startrun.dat file, then restart the switch.

**DATA** This directory is for saving log.dat file which records alarm information.

**Note:**

If IMG, CFG or DATA is unavailable in FLASH, create them manually with **mkdir** command.

# Operating File System Management

ZXR10 8900 series switch provides many commands for file operations. Command format is similar to DOS commands as present in Microsoft Windows Operating System.

To configure file system management, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**copy** <*source-device*><*source-file*><*destination-device*><*destination-file*> | This copies files between Flash and FTP/TFTP server |
| 2 | ZXR10#**pwd** | This displays current directory path |
| 3 | ZXR10#**dir** [<*directory*>] | This displays files, subdirectory information under a designated directory |
| 4 | ZXR10#**delete** <*filename*> | This deletes the files under the a designated directory of the current device |
| 5 | ZXR10#**cd** <*directory*> | This enables to enter specified directory or the current device |
| 6 | ZXR10#**cd..** | This returns to the superior directory |
| 7 | ZXR10#**mkdir** <*directory*> | This creates new directory in flash |
| 8 | ZXR10#**rmdir** <*directory-name*> | This deletes designated directory from flash |
| 9 | ZXR10#**rename** <*source-filename*><*destination-filename*> | This modifies the name of the designated file or directory in a flash |

**Result:** File system management has been configured.

**Example** This example shows how to view the current files in the Flash.

```
ZXR10#dir
Directory of flash:/
       attribute    size       date         time  name
   1     drwx       512     MAY-17-2004  14:22:10 IMG
   2     drwx       512     MAY-17-2004  14:38:22 CFG
   3     drwx       512     MAY-17-2004  14:38:22 DATA
65007616 bytes total (48863232 bytes free)
ZXR10#cd img
ZXR10#dir
Directory of flash:/img
       attribute    size       date         time  name
   1     drwx       512     MAY-17-2004  14:22:10 .
   2     drwx       512     MAY-17-2004  14:22:10 ..
   3     -rwx  15922273    MAY-17-2004  14:29:18 ZXR10.ZAR
65007616 bytes total (48863232 bytes free)
ZXR10#
```

**Example** This example shows how to create a directory ABC in the Flash and then delete it.

```
ZXR10#mkdir ABC
/*Add a subdirectory ABC under the current directory*/

ZXR10#dir
/*Check the current directory information and the directory ABC
can be successfully added*/

Directory of flash:/
       attribute    size       date         time  name
   1     drwx       512     MAY-17-2004  14:22:10 IMG
   2     drwx       512     MAY-17-2004  14:38:22 CFG
   3     drwx       512     MAY-17-2004  14:38:22 DATA
   4     drwx       512     MAY-17-2004  15:40:24 ABC
65007616 bytes total (48861184 bytes free)

ZXR10#rmdir ABC
/*Delete the subdirectory ABC*/

ZXR10#dir
/*Check the current directory information and the directory ABC
has been deleted successfully)

Directory of flash:/
       attribute    size       date         time  name
   1     drwx       512     MAY-17-2004  14:22:10 IMG
   2     drwx       512     MAY-17-2004  14:38:22 CFG
   3     drwx       512     MAY-17-2004  14:38:22 DATA
65007616 bytes total (48863232 bytes free)

ZXR10#
```

# FTP/TFTP Connection Configuration

ZXR10 8900 series switch serves as the client terminal of FTP/TFTP. It is possible to take files backup and to restore them. On ZXR10 8900 series switch, configuration can be imported by FTP/TFTP.

# Configuring a Switch as FTP Client Terminal

**Prerequisites** Enable FTP server software in the background host and switch communicates as client terminal.

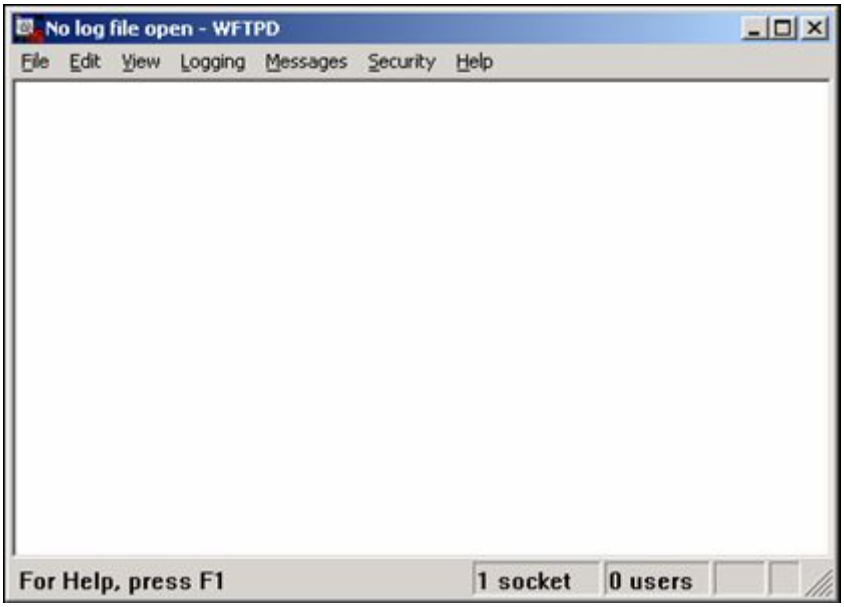
**Context** To configure switch serving as FTP client terminal, perform the following steps.

**Steps**

1. Run WFTPD software in the background host.

   A window appears, as shown in Figure 10.

**FIGURE 10 WFTPD WINDOW**




2. Click **Security**, select **User/Rights...**, and perform the following operations.

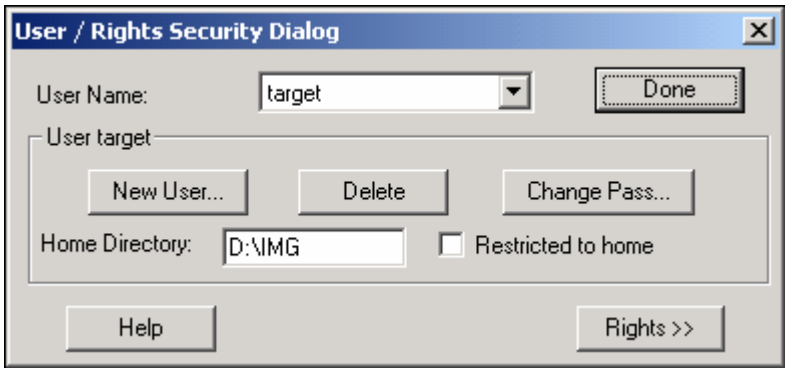
   i. Click **New Use...** to create a new user, such as target, with password enabled.

   ii. Select user name **target** in the drop-down list of **User Name**.

   iii. Input the directory saving version files or configuration files in the **Home Directory** box, such as D: \IMG.

   After configuration is completed, a dialog box appears, as shown in Figure 11.

**FIGURE 11 USER/RIGHTS SECURITY DIALOG BOX**




3. Click **Done** to complete the settings.

**END OF STEPS**

**Result** FTP client is configured. After enabling FTP server, execute copy command in the switch to back up/restore file and import/export configuration.

# Configuring a Switch as TFTP Client Terminal

**Prerequisites** Enable TFTP server software in the background host and switch communication as client terminal.

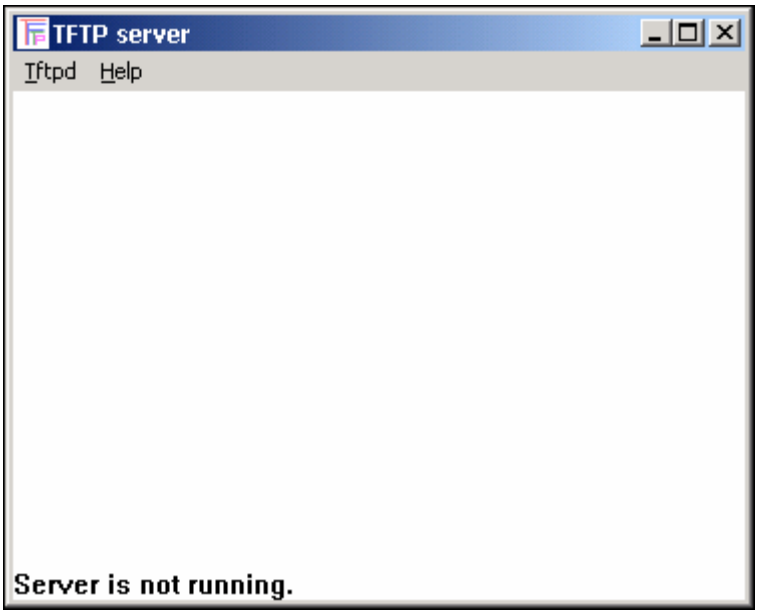
**Context** To configure a switch serving as TFTP client terminal, perform the following steps.

**Steps**

1. Run TFTPD software in the background host.

   A window appears, as shown in Figure 12.

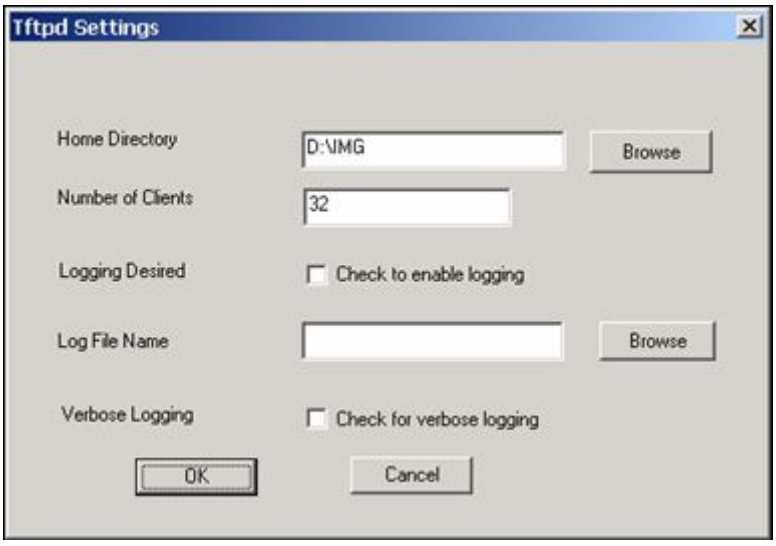
**FIGURE 12 TFTPD WINDOW**




2. Click **Tftpd > Configure**. Adialog box appears. Click **Browse**, and select the file saving version files or configuration files, such as D:\IMG.

   After configuration is completed, a dialog box appears, as shown in Figure 13.

**FIGURE 13 CONFIGURATION DIALOG BOX**




3. Click **OK** to complete setting.

**END OF STEPS**

**Result** TFTP client is configured. After enabling TFTP server, execute copy command in the switch to back up/restore file and import/export configuration.

# File Backup and Restoration

## Backing up Configuration File

After saving the configuration file to startrun.dat with **write** command, users can back up the file to background FTP/TFTP server to prevent the file from being destroyed.

To back up the configuration file, use the following command.

| Command | Function |
|---|---|
| ZXR10#**copy** *<source-device><source-file><destination-device><destination-file>* | This backs up configuration file |

**Example** This example shows **copy** command that takes a backup of configuration files in FLASH to background TFTP server.

```
ZXR10#copy flash: /cfg/startrun.dat tftp: //168.1.1.1
/startrun.dat
```

## Restoring Configuration File

To restore configuration files, use the following command.

| Command | Function |
|---|---|
| ZXR10#**copy** *<source-device><source-file><destination-device><destination-file>* | This restores configuration files |

**Example** This example shows **copy** command that restores backup configuration files from background TFTP server.

```
ZXR10#copy tftp: //168.1.1.1/startrun.dat flash:
/cfg/startrun.dat
```

## Backing up System Software Version

Before users upgrade software version, it is necessary to take a backup of the running version files up to background server. If the system fails to load new version, users can restore the old version from the background server. Software version file backup is similar to configuration file backup.

To back up version files, use the following command.

| Command | Function |
|---|---|
| ZXR10#**copy** *<source-device><source-file><destination-device><destination-file>* | This backs up version files |

**Example** This example shows **copy** command that takes a backup of the software version file in FLASH to directory IMG in root directory of background TFTP server.

```
ZXR10#copy flash: /img/zxr10.zar tftp: //
168.1.1.1/img/zxr10.zar
```

## Restoring System Software Version

Purpose of version restoration is to re-transmit the backup software version file in background server through FTP/TFTP to FLASH in foreground switch. It is important to perform restoration operation when version upgrade is failed.

**Note:**

Version restoration and version upgrade procedures are almost the same, please refer to Software Version Upgrade.

# Ststem Software Version Upgrade

Software version upgrade is only made when the original version fails to support certain functions. Improper operation may lead to upgrade failure and system booting failure. Therefore, before starting to upgrade the version, read related documents to understand principle, operation and upgrade procedure of the ZXR10 8900 series switch.

## Upgrading Version at Abnormality

**Prerequisites** The following requirements are to be completed before users begin software version upgrade.

- Connect the configuration port (Console port of MP board) of ZXR10 8900 series switch to the serial interface of background host by configuration cable delivered with the product. Connect management Ethernet interface of the device (10/100M Ethernet interface) to network interface of background host by

straight-through Ethernet cable. Make sure that both interfaces are connected in a proper way.

- Start the background FTP server.

**Context** To upgrade the version at abnormality, perform the following steps.

**Steps**

1. Start ZXR10 8900 series switch using HyperTerminal and press any key to enter Boot status.

   The following content appears.

```
ZXR10 System Boot Version: 1.0
Creation date: Dec 31 2002, 14:01:52
(Omitted)
Press any key to stop for change parameters...
2
 [ZXR10 Boot]:
```

2. Input "c" in Boot status. Enter parameter modification status after inputting an Enter.

   i. Change the boot mode to boot from background FTP.

   ii. Change the FTP server address to the corresponding background host address.

   iii. Change the client terminal address and gateway address to switch administrative Ethernet interface address.

   iv. Set corresponding subnet mask and FTP username and password.

   [ZXR10 Boot] prompt appears after above parameter modification is completed.

```
[ZXR10 Boot]:c
' .'  = clear field;  ' -'  = go to previous field;  ^D = quit
Boot Location [0:Net,1:Flash] : 0
(0 means booting from background FTP;
1 means booting from FLASH)
Client IP [0:bootp]: 168.4.168.168
(Corresponds to administrative Ethernet port address)
Netmask: 255.255.0.0
Server IP [0:bootp]: 168.4.168.89
(Corresponds to background FTP server address)
Gateway IP: 168.4.168.168
(Corresponds to administrative Ethernet port address)
FTP User: target (Corresponds to FTP username target)
FTP Password:    (Corresponds to target user password)
FTP Password Confirm:
Boot Path: zxr10.zar     (Use default)
Enable Password:         (Use default)
Enable Password Confirm: (Use default)
[ZXR10 Boot]:
```

3. Input "@". System boots the version from background FTP server automatically after carriage return.

   The following information is displayed.

```
[ZXR10 Boot]:@
Loading...  get file zxr10.zar[15922273] successfully!
file size 15922273.
(Omitted)

******************************************************
Welcome to ZXR10 10G Routing switch of ZTE Corporation
******************************************************
ZXR10>
```

4. If system has been started normally, use **show version** command to check whether the new version is running in the memory or not. If it is the old running version, it indicates that

booting from background server failed, in this case repeat the operations from step 1.

5. Delete the old version file zxr10.zar in the directory IMG in FLASH with **delete** command. Old version file is renamed for backup due to of space in FLASH is sufficient.
6. Copy the new version file in background FTP server to IMG directory in FLASH. Version file name is zxr10.zar.

   The following information is displayed.

```
ZXR10#copy ftp: mng //168.4.168.89/zxr10.zar@target:target
flash: /img/zxr10.zar
Starting copying file


file copying successful.
ZXR10#
```

**Note:**

If copying version files from the management Ethernet of MP board, in the **copy** command, **ftp** must be followed with **mng**.

7. Check whether new version file is available in FLASH or not. If the new version file is unavailable, it indicates the file copy failure, please execute step 6 to re-copy the version.
8. Restart ZXR10 8900 series switch and follow the methods in step 4, and boot the system from FLASH enabled, at this time, "Boot path" is changed into"/flash/img/zxr10.zar automatically.

**Note:**

Boot mode is changed to boot from FLASH by using **nvram imgfile-location local** command in global configuration mode.

9. Input "@" in [ZXR10 Boot]: now system will boot a new version from FLASH after carriage return.
10. After a normal boot-up, check the running version to confirm the successful upgrade.

**END OF STEPS**

**Result** The version has been updated at abnormality.

# Upgrading Version at Normality

**Prerequisites** The following requirements are to be completed before users begin software version upgrade.

- Connect the configuration port (Console port of MP board) of ZXR10 8900 series switch to the serial interface of background

host by configuration cable delivered with the product. Connect management Ethernet interface of the device (10/100M Ethernet interface) to network interface of background host by straight-through Ethernet cable. Make sure that both interfaces are connected properly.

- IP addresses of background host for upgrade and management Ethernet interface on the device are set to the same network segment. Make sure that the background host could ping to the management Ethernet interface successfully.
- Start the background FTP server.

**Context** To upgrade the version at normality, perform the following steps.

**Steps**

1. View the information of the running version.
2. Delete the old version file in the directory IMG in FLASH with **delete** command. The old version file can be renamed if there is sufficient space in FLASH.
3. Copy the new version file in background FTP server to IMG directory in FLASH. Version file name is zxr10.zar.
4. Check whether the new version file is available in directory IMG in FLASH. If the new version file is unavailable, it indicates the copy failure, please execute step 3 to recopy the version.
5. After a normal switch boot-up, check the running version to confirm whether the upgrade is successful or not.

**END OF STEPS**

**Result** The version has been updated at normality.

# Upgrading Version without Interrupting System

**Prerequisites** The following requirements are to be completed before users begin software version upgrade.

- Connect the configuration port (Console port of MP board) of ZXR10 8900 series switch to the serial interface of background host by configuration cable delivered with the product. Connect management Ethernet interface of the device (10/100M Ethernet interface) to network interface of background host by straight-through Ethernet cable. Make sure that both interfaces are connected in a proper way.
- IP addresses of background host for upgrade and management Ethernet interface on the device are set to the same network segment.
- Start the background FTP server.

**Context** When the users want to update the version without interrupting the system, users can update the version through the secondary controlled switch board first, and then switch over the primary controlled switch board and the secondary controlled switch board. After that, the users update the new secondary controlled switch

board. The line interface cards should be rebooted after the version update.

To update the version without interrupting the system, perform the following steps.

**Steps**

1. View the information of the current version.
2. Delete the old version file in the directory IMG in FLASH with **delete** command. The old version file can be renamed if there is sufficient space in FLASH.
3. Copy the new version file in background FTP server to IMG directory in FLASH. Version file name is zxr10.zar.
4. Check whether the new version file is available in directory IMG in FLASH. If the new version file is unavailable, it indicates the copy failure, please execute step 3 to recopy the version.
5. Copy the new version file in the directory IMG in FLASH to memory with **update-imgfile** command.
6. Reboot the secondary board with **reload mp slave** command.
7. Switch over the primary board and secondary card with **redundancy force** command.
8. To reboot the interface cards one by one with **reload slot** <*board unit number*> command.
9. Check the running version to confirm whether the upgrade is successful or not.

**END OF STEPS**

**Result** The version has been updated without interrupting the system.

# System Parameter Configuration

## Configuring a Hostname

To set a hostname of system, use the following command.

| Command | Function |
|---|---|
| `ZXR10(config)#`**hostname** <*network-name*> | This sets hostname of system |

**Note:**

By default, the system hostname is ZXR10, which can be modified with the hostname command in the global configuration mode. Log on to router again after hostname modification and the prompt will include the new hostname.

# Configuring a Welcome Message

To set welcome message upon system boot or when login on telnet, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**banner incoming** | This sets the greeting words |

**Example** This example shows how to configure welcome message upon system boot.

```
ZXR10(config)#banner incoming #
Enter TEXT message.  End with the character ' #' .
**************************************
    Welcome to ZXR10 Router World
**************************************
#
ZXR10(config)#
```

# Configuring a Password of Privileged Mode

To prevent an unauthorized user from modifying the configuration, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**enable secret** {**0** *<password>*\|**5** *<password>*\|*<password>*} | This sets password |

# Configuring Telnet Username and Password

To set Telnet username and password, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**username** <*username*> **password** <*password*> | This sets Telnet user and password |

# Configuring System Time

To set system time, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**clock set** <*current-time*><*month*><*day*><*year*> | This sets system time |

# Configuring Version Load Selection

When users upgrade switch versions, the old version files are usually kept in case of upgrade failure. The operation steps are described below.

1. Modify the name of old version file.
2. Upload new version file to the switch.
3. Reboot the switch.

All version files are saved in the same directory. Version file loaded normally are named ZXR10.ZAR. When users are upgrading multiple switches, or when there are multiple version files in a switch, the users who perform usual upgrade steps likely feel confused. Besides, users have to compare the memories that the version files take, which is inconvenient.

When version file is uploading to flash, users can specify the directory and name of version file, and then select the needed version file when booting the switch. This is the function that version load selection module provides. When device is running normally, users can configure the version file name and directory to load when the device is rebooted next time.

To configure version load selection function, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**nvram imgfile-location** {**local** {**flash** \| **sd**}<*filename*>}\| **network** <*filename*>} | This configures location of image file |

Parameter descriptions:

| Parameter | Description |
|---|---|
| **local** | Image file is in local device. |

| Parameter | Description |
|---|---|
| **flash** | The type of storage device from which version file is booted is flash. |
| **sd** | The type of storage device from which version file is booted is SD card. |
| **network** | Image file is on a network. |
| *<filename>* | File name, within 80 characters |

The following characters are available in version file name:

0123456789abcdefghijklmnopqrstuvwxyz_ABCDEFGHI-JKLMNOPQRSTUVWXYZ/.;,-=+$#~@%!&[]{}

If version file is configured to boot from network, file name can contain path in designated FTP directory. For example, the designated FTP directory is sysm, a user has entered nets in sysm directory, the version file name can contain path in nets directory.

The command to configure version load selection function can be used together with **nvram boot-password**, **nvram boot-server**, **nvram boot-username** and **nvram default-gateway** commands.

**Example** This example shows how to configure booting from local device

```
ZXR10(config)#nvram imgfile-location local
```

This example shows how to configure booting from network.

```
ZXR10(config)#nvram imgfile-location network sys.img
```

# Saving Command Log File

A switch can save some log files. However, after a switch is rebooted, the log files before rebooting will be lost. If log files are saved to flash or SD card, they will not be lost after switch is rebooted. The switch provides the function that log files can be saved and synchronized to flash and SD card. Storage path, file name and size can be configured. The size of file ranges from 64K bytes to 1024K bytes. By default, it is 256K bytes. When the size exceeds the maximum size, the earliest parts of logs are deleted.

**Note:**

By default, the file is saved in flash/data directory, and file name is logfile.txt.

To save command log file, use the following command.

| Command | Function |
|---|---|
| ZXR10#**write cmdlog** {**flash** \| **sd**}[**start-time** <*date*><*time*>][**end-time** <*date*><*time*>][**filename** <*filepath/file*>] | This saves the contents in command log buffer as a file. The file is saved in flash/data directory. |

Parameter descriptions:

| Parameter | Description |
|---|---|
| **start-time** <*date*><*time*> | The starting time when alarms begin to be recorded. By default, it is the time of the earliest alarm log in current alarm buffer. |
| **end-time** <*date*><*time*> | The time when alarm occurs. By default, it is the time of the latest alarm log in current alarm buffer. |
| **flash** | Command log file is saved to flash. |
| **sd** | Log file is saved to SD card. By default, it is saved to flash. |
| **filename** <*filepath/file*> | The path and name of log file, within 32 characters. By default, the path and name is /data/cmd.log. |

# Configuring Saving Time of Alarm Log

Event information is kept in system buffer of a switch. When the buffer is full, system clears the earliest event information. If saving time is configured, system clears corresponding events automatically when it is time. When there are a lot of events and buffer is full before saving time comes, events are cleared according to configuration of logging buffer clearing. Error of saving time is within 1 minute. Saving time can be 0 or a value in the range of 30 to 65335 minutes. By default, it is 0, indicating that system clears events according to configuration of logging buffer clearing when buffer is full.

To configure saving time of alarm log, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**write alarmlog** {**flash** \| **sd**}[**start-time** <*date*><*time*>][**end-time** <*date*><*time*>][**filename** <*filepath/file*>] | This saves contents in alarm log buffer in designated file form on other devices |

Parameter descriptions:

| Parameter | Description |
|---|---|
| **flash** | Alarm log file is saved to flash. |
| **sd** | Alarm log file is saved to SD card. |
| **start-time** <*date*><*time*> | The starting time of alarm to be recorded that occurs earliest. |
| **end-time** <*date*><*time*> | The starting time of alarm to be recorded that occurs latest. |
| **filename** <*filepath/file*> | The path and name of log file, within 32 characters. By default, the path and name is /data/cmd.log. |

**Example** This example shows how to save alarm log to flash/data/alarm.log.

```
ZXR10(config)# write alarmlog flash start-time
6-12-2008 00:00:01 end-time 6-12-2008 23:59:59
```

This example shows how to save alarm log to flash/aaa.log.

```
ZXR10(config)# write alarmlog flash start-time
06-25-2008 15:03:00 end-time 06-25-2008 15:04:45 filename aaa.log
```

# System Information View

System information view includes the following topics.

## Viewing Hardware and Software Versions

To view hardware and software versions of the system, use the following command.

| Command | Function |
|---|---|
| ZXR10#**show version** | This displays the version information about the software and hardware of system |

## Viewing Current Running Configuration Information

To view running configuration, use the following command.

| Command | Function |
|---|---|
| ZXR10#**show running-config** | This displays the running configuration |

# Viewing CPU Information

To view CPU information, use the following command.

| Command | Function |
|---|---|
| ZXR10#**show process** | This displays CPU information |

# Viewing Boot Information of Current Running Board

To view boot information of current running board, use the following command.

| Command | Function |
|---|---|
| ZXR10#**show boot** | This displays boot information of current running board |

**Example** This example shows how to view boot information of current running board.

```
ZXR10#show boot
[MEC2, panel 1, master]
Bootrom Version : V1.84
Creation Date   : 2008/6/17
Update Support  : YES

[MEC2, panel 2, slave]
Bootrom Version : V1.84
Creation Date   : 2008/6/17
Update Support  : YES

[NPCI, panel 12]
Bootrom Version : V1.83
Creation Date   : 2008/7/6
Update Support  : YES
```

# Viewing System Diagnosis Information

When malfunction occurs on network, it is required to collect diagnosis information as soon as possible and solve the problem. It is an urgent task to analyze the malfunction, and usually some important information is not collected. ZXR10 8900 series switch

provides function to collect and save diagnosis information. The directory and name of saved file can be configured. By default, the file directory is flash/user and is named diag-info.txt.

Diagnosis information includes the following contents:

- Current time
- Current version, as well as configuration of boards and cards
- Current configuration
- Displaying log
- Interface configurations
- State of link aggregation groups
- VLAN configuration
- MAC table configuration
- ARP configuration
- Current routing table
- The latest 50 times of operations of FIB table
- IP traffic information
- Detailed memory usage information
- CPU usage ratio
- Process information
- Queue information
- IGMP snooping information
- IP multicast routing table
- Layer 3 multicast joining information
- IP multicast forwarding table
- File information in flash
- Detailed information of software abnormity
- Resetting information of main control board
- Changeover information of active and standby boards
- Abnormal information of main control board intermitting
- Software resetting information of line interface card
- Abnormal information of line interface card intermitting
- Spanning tree state on port
- Protocol VLAN information
- Selective QinQ information
- MPLS/VPN LDP information
- MPLS/VPN LSP information
- VPN routing information
- QoS information

To view system diagnosis information, use the following command.

| Command | Function |
|---|---|
| ZXR10#**show diagnostic information**[{[**detail**[{[**module** <*module-name*>[|{**begin** \| **exclude** \| **include**}]][|{**begin** \| **exclude** \| **include**}]}]]|[**module** <*module-name*>[|{**begin** \| **exclude** \| **include**}]]|[**save**]}] | This displays information of the whole system for malfunction analysis when malfunction occurs in the system or a module |

By default, there is no parameter and brief system information is displayed page by page. The displayed information is not saved by default.

Parameter descriptions:

| Parameter | Description |
|---|---|
| **detail** | Display detailed system information. |
| **module** <*module-name*> | Display information of designated module. |
| **begin** | Display configuration information beginning with designated character or character string. |
| **exclude** | Display configuration information excluding designated character or character string. |
| **include** | Display configuration information including designated character or character string. |
| **save** | Save current system information to flash. |

Chapter 4

# CLI Privilege Classification

**Table of Contents**

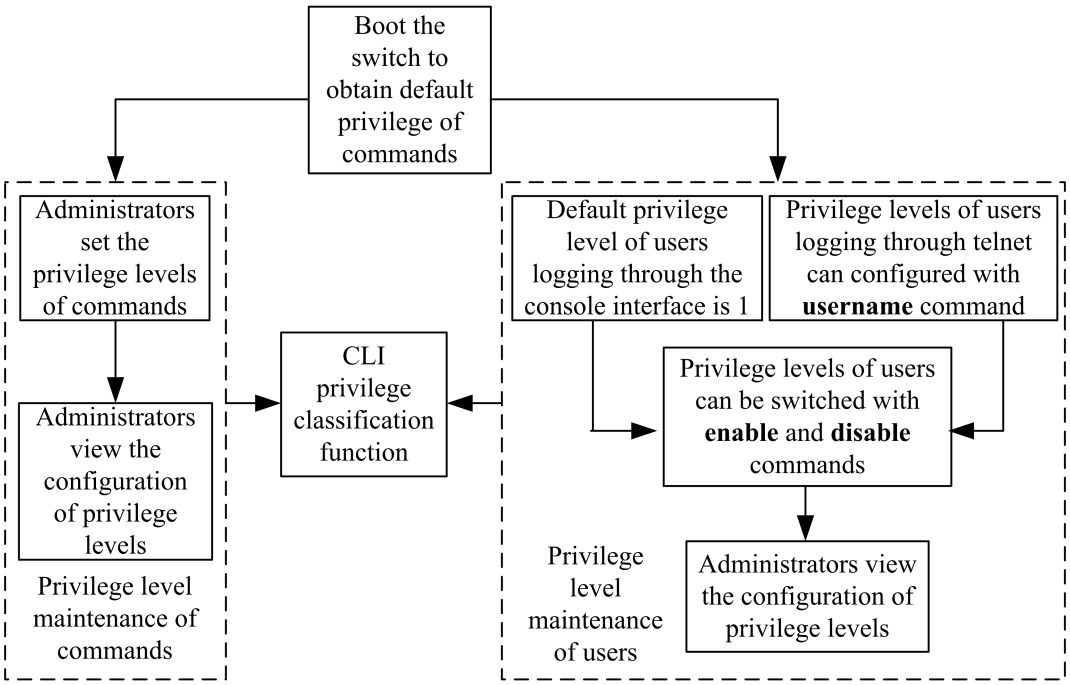
CLI Privilege Classification Overview .....37
Configuring CLI Privilege Classification .....38
CLI Privilege Classification Configuration Example .....42
Maintenance and Diagnosis of CLI Privilege Classification .....42

## CLI Privilege Classification Overview

ZXR10 8900 series switch supports CLI privilege classification function. There are 16 levels. Different users can have different privilege levels. The higher privilege level users have, the more commands users can use. The administrators have the highest level (Level 15). Therefore, they can set the levels of different commands.

CLI privilege classification function consists of two parts: privilege level maintenance of commands and users, as shown in Figure 14.

**FIGURE 14 CLI PRIVILEGE CLASSIFICATION FUNCTION**


Boot the switch to obtain default privilege of commands
Administrators set the privilege levels of commands
Administrators view the configuration of privilege levels
Privilege level maintenance of commands
CLI privilege classification function
Default privilege level of users logging through the console interface is 1
Privilege levels of users logging through telnet can configured with username command
Privilege levels of users can be switched with enable and disable commands
Privilege level maintenance of users
Administrators view the configuration of privilege levels

**Privilege Level Maintenance of Commands**

When a device is booted, each command has a default privilege level. Administrators can modify the privilege levels of the commands.

**Privilege Level Maintenance of Users**

Administrators also can modify the privilege levels of the users who log into the switch. When a user's privilege level is the same with or higher than the privilege level of a command, the user can use the command.

# Configuring CLI Privilege Classification

## Configuring Telnet User

Considering security, the privilege level of a user only can be configured by the administrators. That is, after a user logs in to the switch, the user can not modify own login password and privilege level. Administrators do not need to check the password when modifying the privilege level of the user.

To configure the privilege level of a telnet login user, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**username** <*username*> **password** <*password*> **privilege** <*level*> | This configures the user name, password and privilege level of a telnet login user |

**Note:**

To delete the user, use **no username** <*username*> command.

**Example** This example shows how to configure the privilege level to 12 of a user named test.

```
ZXR10(config)#username test password test privilege 12
```

When the user telnets to log in to the switch, the prompt is shown below.

```
Username:test
Password:
ZXR10#
```

**Example** This example shows hot to change the privilege level to 1 of the user.

```
ZXR10(config)#username test password test privilege 1
```

When the user telnets to log in to the switch, the prompt is shown below.

```
Username:test
Password:
ZXR10>
```

**Note:**

When a user with privilege level 2~15 logs in to the switch, the prompt is "#". When a user with privilege level 1 logs in to the switch, the prompt is ">", indicating that user should input the enabling password, as shown below.

```
Username:test
Password:
ZXR10#enable 12
//if no parameter is input after enable,
the default privilege level is 15
Password:
ZXR10#
```

# Configuring an Enabling Password

Administrators can configure an enabling password for each privilege level. When a user with lower privilege level wants to obtain a higher privilege level, the user should input the enabling password.

To configure an enabling password for a privilege level, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**enable secret level** <*level*><*password*> | This configures an enabling password for a privilege level |

**Note:**

To delete the enabling password, use **no enable secret level** <*level*> command.

**Example** This example shows how to configure an enabling password and when to use this password.

Administrators configure the privilege level to 1 for a user named test, as shown below.

```
ZXR10(config)#username test password test privilege 1
```

The enabling password of privilege level 12 is configured to "zte", as shown below.

```
ZXR10(config)#enable secret level 12 zte
```

When the user logs in to the switch and wants to change the privilege level to 12, the user should input the enabling password, as shown below.

```
Username:test
Password:       //this password should be "test"
ZXR10>enable 12
Password:       //this password should be "zte"
ZXR10#
```

# Configuring Privilege Level of a Command

By configuring privilege levels of commands, administrators can control the range of commands that users can use. When the privilege level of a user is higher or equals to the privilege level of a command, the user can use the command. By default, the privilege level of administrators is 15. They can use all commands.

To configure the privilege level of a command, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**privilege** <*logic-mode*>{{**all level**}\|**level**}<*level*><*command-keywords*> | This configures the privilege level of a command |

**Example** This example shows how to configure the privilege level to 12 for all commands beginning with **show interface**.

1. View all commands beginning with show with user privilege level of 12.

```
ZXR10#show ?
privilege  Show current privilege level
```

The result shows that only show privilege command is displayed.

**Note:**

If there is no command with privilege level 12, after the user inputs "?" for help, no command will be displayed.

2. Configure the user privilege level to 15.

```
ZXR10#enable
Password:
ZXR10#
```

3. Configure the privilege level to 12 for all commands beginning with **show interface**.

```
ZXR10#configure terminal
ZXR10(config)#privilege show all level 12 show interface
```

4. Go back to privilege level 12.

```
ZXR10#enable 12
ZXR10#
```

**Note:**

When the user goes back to a lower privilege level from a higher privilege level, the user does not need to input enabling password.

5. View all commands beginning with **show** with user privilege level of 12.

```
ZXR10#show ?
interface  Show interface property and statistics
privilege  Show current privilege level
```

The result shows that **show interface** command is added to commands with privilege level of 12.

Use **show interface** command to view interface information, as shown below.

```
ZXR10#show interface gei_1/2
gei_1/2 is up,  line protocol is up
  Description is none
  The port is electric
  Duplex full
  Mdi type:auto
  VLAN mode is hybrid, pvid 1
  MTU 1500 bytes    BW 1000000 Kbits
  Last clearing of "show interface" counters never
  120 seconds input  rate:   0 Bps,              0 pps
  120 seconds output rate:   5 Bps,              0 pps
  ......
```

# CLI Privilege Classification Configuration Example

Use user privilege level 15 to configure a user named test with privilege level of 10. The configuration is shown below.

```
ZXR10(config)#username test password test privilege 10
ZXR10(config)#enable secret level 10 test123
ZXR10(config)#privilege show all level 10 show run
```

The configuration result is shown below.

```
ZXR10(config)#exit
ZXR10#enable 10
ZXR10#show run
Building configuration...
!
!
urpf log off
!
......
```

# Maintenance and Diagnosis of CLI Privilege Classification

To configure maintenance and diagnosis of CLI privilege classification, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show privilege cur-mode** {**detail** \|{**level** <*level*>}\|{**node** <*command-keywords*>} | This views the privilege level of commands in current mode |
| 2 | ZXR10#**show privilege show-mode** {**detail** \|{**level** <*level*>}\|{**node** <*command-keywords*>} | This views the privilege level of commands in show mode |

Chapter 5

# Port Configuration

**Table of Contents**

Port Basic Configuration ..... 43
Port Mirroring Configuration ..... 52
ERSPAN Configuration ..... 54
Configuring ERSPAN ..... 55
ERSPAN Configuration Example ..... 55
Port Loop Detection Configuration ..... 56

# Port Basic Configuration

## Port Basic Configuration Overview

ZXR10 8900 series switch provides fast Ethernet port, gigabit Ethernet port and 10-gigabit Ethernet port.

- Fast Ethernet electrical interface supports full-duplex/half-duplex, 10/100M and MDI/MDIX self-adaptive function. Default working mode is auto-negotiation. It negotiates work mode and rate with the opposite end devices.
- Gigabit Ethernet electrical interface supports full-duplex/half-duplex, 10/100/1000M and MDI/MDIX self-adaptive function. Default working mode is auto-negotiation. It negotiates working mode and rate with the opposite end devices.
- Gigabit Ethernet electrical interface works in gigabit full-duplex mode. Duplex mode and rate of the port cannot be configured but auto-negotiation mode can be configured.
- 10 gigabit Ethernet optical interface works in 10 gigabit full-duplex mode. Auto-negotiation, duplex mode and rate of the port cannot be configured.

System adds the ports automatically: user plugs interface board into the corresponding slot, when the interface board starts normally, port of the interface board has been added to the system port list automatically.

**Port Naming Rules**

ZXR10 8900 series switch names the ports in the following way:

Port type_Slot **No.**/Port **No.**

- Port type covers:

  FEI: Fast Ethernet Interface

GEI: Gigabit Ethernet Interface

XGEI: 10 Gigabit Ethernet Interface

- Slot **No.**

  ZXR10 8908 provides 10 plug-in slots that are numbered from top to down, where No. 5 and No. 6 are MP plug-in slots and rest are the interface board module plug-in slots.

- Port **No.**

  Interface board ports number starts from 1.

  fei_2/8 means the eighth port in the No. 2 slot fast Ethernet interface board.

  gei_6/1 means the first port in the No. 6 slot gigabit Ethernet interface board.

  xgei_7/2 means the second port in the No. 7 slot 10 gigabit Ethernet interface board.

# Enabling an Ethernet Port

To enable an Ethernet port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | `ZXR10(config-if)#`**no shutdown** | This enables an Ethernet port |
| 3 | `ZXR10(config-if)#`**byname** <*by-name*> | This sets port byname |

**Note:**

- To disable an Ethernet port, use **shutdown** command.
- The **shutdown** command makes the physical link status of the port change into **down** and the link LED of the port go dark. All ports are open by default.
- Port byname is to distinguish the ports for easier memorization. It is possible to replace the port name with **byname** command when users perform operation over the port.

# Enabling Auto-Negotiation

To enable auto-negotiation function of an interface, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**negotiation auto** | This enables Ethernet port auto-negotiation |

**Note:**

- To disable auto-negotiation function of an interface, use **no negotiation auto** command.
- 10 gigabit Ethernet optical interface does not support auto-negotiation. It is fixed to work in 10 gigabit full-duplex mode.

# Configuring Duplex Mode

To configure Ethernet port duplex mode, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**duplex** {**half**\|**full**} | This configures Ethernet port duplex mode |

**Note:**

Only the Ethernet electrical interface can be configured with duplex mode. Before configuring the Ethernet port duplex mode, disable auto-negotiation function first.

# Configuring Ethernet Port Rate

To configure Ethernet port rate, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**speed** {**10**\|**100**\|**1000**} | This configures Ethernet port speed |

**Note:**

Only the Ethernet electrical interface can be configured with port rate. Before configuring the port rate, disable auto-negotiation function first.

# Configuring Traffic Control

To configure Ethernet port traffic control, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**flowcontrol** {**enable**\|**disable**} | This configures Ethernet port flow control |

**Note:**

Ethernet port uses traffic control to restrain the packets sent to the port in a period of time. When the receiving buffer is full, a port sends a "pause" packet notifying the remote port to suspend packet transmission for a period of time. Ethernet port can also receive "pause" packet from other devices, and execute operations according to the packet regulation.

# Allowing Jumbo-Frame

To allow jumbo-frame to pass the Ethernet port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**jumbo-frame enable** | This allows jumbo-frame to pass the Ethernet port |

**Note:**

- By default, the maximum allowed length of the frame passing Ethernet port is 1560 bytes, and jumbo frame is prohibited from passing. When jumbo frame is allowed, the maximum allowed length is 9216 bytes.
- To prohibit jumbo-frame to pass the Ethernet port, use **jumbo-frame disable** command.

# Configuring Broadcast Storm Suppression

To configure Ethernet port broadcast storm suppression, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**broadcast-limit** {{**percent** <*percent*>}\|{**value** <*value*>}} | This configures Ethernet port broadcast storm suppression |

**Note:**

- It is possible to limit the volume of broadcast flow that is allowed to pass through the Ethernet port. System discards the broadcast flow exceeding the set value to lower the rate of broadcast flow to a reasonable range. It suppresses broadcast storm and avoids network congestion, ensuring normal operation of network service.
- Broadcast storm suppression ratio takes the line speed percentage of maximum flow as the parameter. If percentage is lower then allowed broadcast flow is smaller as well. 100% means that the broadcast storm passing through the port is not suppressed.

# Configuring Multicast Suppression

To configure multicast suppression of Ethernet port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**multicast-limit** {{**percent** <*percent*>}\|{**value** <*value*>}} | This configures multicast suppression of Ethernet port |

# Configuring Unknown Unicast Suppression

To configure unknown unicast suppression of Ethernet port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**unknowcast-limit** {{**percent** <*percent*>}\|{**value** <*value*>}} | This configures unknown unicast suppression of Ethernet port |

# Enabling Fast Port Detection Function

To enable fast port detection function, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**zfid interface** <*port-list*> | This enables fast port detection function |

**Note:**

This function detects the change of the status on an interface (for example, from **up** to **down**), and informs protocols such as ZESR, ZESS and link aggregation of the change to speed up the running of the protocols. As the function costs resource, it is recommended to enable the function only on related ports.

# Configuring FEFI Function

To configure FEFI function, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**fefi** {**enable** \| **disable**} | This configures FEFI function |

# Configuring TCP Rate Limit

To configure TCP rate limit, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**tcp-syn protect rate-limit** <*64-1000000*> | This configures TCP rate limit |

# Configuring Switch of Optical or Electrical Port

To switch optical or electrical port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** {<*port-name*>\|**byname** <*by-name*>} | This accesses port configuration mode |
| 2 | ZXR10(config-if)#**hybrid-attribute** {**copper** \| **fiber**} | This switches optical or electrical port |

**Note:**

This command only can not be used on purely optical or electrical interfaces.

# Viewing Port Information

To view port information, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**show interface** [<*port-name*>] | This views status information of Ethernet port |
| 2 | ZXR10(config)#**show zfid** [**interface** <*port-list*>] | This views information on port that enables fast port detection function |
| 3 | ZXR10(config)#**show linkage-group** [**id**] | This views linkage configuration information on a port |
| 4 | ZXR10(config)#**show running-config interface** <*port-name*> | This views configuration information of Ethernet port |

To clear port statistical information, use clear counter command.

**Example** This example shows how to view status and statistic information of port gei_2/1.

```
ZXR10(config)#show interface gei_2/1
gei_2/1 is down,  line protocol is down
  Description is none
  Keepalive set:10 sec
  The port is electric
  Duplex half
  Mdi type:auto
vlan mode is access, pvid 2
Vrpf All Discard Count:0      BW 1000000 Kbits
  Last clearing of "show interface" counters never
  120 seconds input  rate      0 Bps,          0 pps
  120 seconds output rate      0 Bps,          0 pps
  Interface peak rate  : input   0 Bps, output   0 Bps
  Interface utilization: input   0%, output    0%
/* Statistic of input/output transmit message,
including statistic of error message */

  Input:
   Packets  : 338                              Bytes: 41572
   Unicasts : 0          Multicasts: 328       Broadcasts: 10
   Undersize: 0          Oversize  : 0         CRC-ERROR : 0
   Dropped  : 0          Fragments : 0         Jabber    : 0
   MacRxErr : 0
  Output:
   Packets  : 1017                             Bytes: 125470
   Unicasts : 0          Multicasts: 1017      Broadcasts: 0
   Collision: 0                                LateCollision: 0

  Total:
   64B      : 20         65-127B   : 975       128-255B  : 360
   256-511B : 0         512-1023B : 0         1024-1518B: 0
ZXR10#
```

**Example** This example shows how to view configuration information of port fei_2/4.

```
ZXR10(config)#show running-config interface fei_2/4
Building configuration...
interface fei_2/4
  negotiation auto
  broadcast-limit 10
  switchport access vlan 1
  switchport qinq normal
ZXR10(config)#
```

# Diagnosing and Testing Link

ZXR10 8900 series switch supports cable line diagnosis analysis test function that detects the line abnormality or line connection abnormality. This test locates the exact position of cable fault, facilitating network management and locating fault.

Both fast Ethernet electrical interface and gigabit Ethernet electrical interface are connected to other devices by network wire. There are four pairs of twisted pair cables in the network wire, in which, fast Ethernet electrical interface uses 1-2 and 3-6 twisted pair cables, gigabit Ethernet electrical interface uses all the four pairs of twisted pair cables including 1-2, 3-6, 4-5 and 7-8. Line detection can detect the status of twisted pair cable. This is described in the following list:

- Open: Open circuit
- Short: Short circuit
- Mismatch: Circuit impedance mismatched
- Good: The circuit is in good condition
- Broken: the circuit is open or short
- Unknown: The result is unknown or undetected
- Fail: Detection failed

If the circuit is faulty, test result outputs the circuit fault location. If the circuit is in good condition, approximate length of the normal circuit is generated.

To diagnose and test link, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**show vct interface** <*port-name*> | This diagnoses and tests link |

**Note:**

Related ports are restarted when line diagnosis analysis test is used. Link will disconnect and then becomes normal. It is usually to test the faulty ports. Be careful when the port is connected with users.

**Example** This example shows how to detect like of port gei_3/1

```
ZXR10(config)#show vct interface gei_3/1
CableStatus    Fault
Pair           1-2       3-6       4-5       7-8
Status         Open      Open      Good      Good
Length         4m        4m        <50m      <50m
ZXR10(config)#
```

# Port Mirroring Configuration

## Port Mirroring Overview

Port mirroring function copies the data of one or more ports (mirrored ports) in the switch to a designated port (monitoring port). It can retrieve the data of mirrored port in the monitoring port by mirroring. Through which it can perform network flow analysis, and error diagnosis.

Port mirroring function on ZXR10 8900 series switch complies with the following rules:

- It supports up to 8 groups of port mirroring, each can support up to 8 mirrored ports.
- In one interface board, one group of port mirroring can be configured at maximum.
- Supports cross-interface-board port mirroring, for example, mirrored port and the monitoring port can be in different interface boards, here, the switch can be configured with one port mirroring at most.
- Monitor the data transmitted or received by the mirrored port only.

## Configuring Port Mirroring

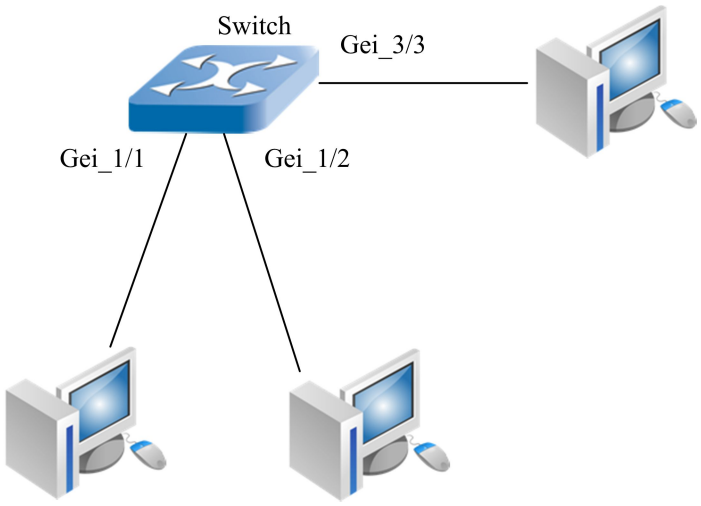
To configure port mirroring, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**monitor session** <*session-number*> | This creates a session |
| 2 | `ZXR10(config-if)#`**monitor session** <*session-number*> **source** [**direction** {**both**\|**cpu-rx**\|**cup-tx**\|**tx**\|**rx**}] | This sets mirrored port |
| 3 | `ZXR10(config-if)#`**monitor session** <*session-number*> **destination** | This sets monitoring port |
| 4 | `ZXR10(confi)#`**show monitor session** {**all**\|<*session -number*>} | This views configuration and status of port mirroring |

## Port Mirroring Configuration Example

As shown in Figure 15, port gei_3/3 is connected with a monitoring computer.

**FIGURE 15 PORT MIRRORING CONFIGURATION EXAMPLE**




To the data received by gei_1/1, as well as the data received and transmitted by gei_1/2, the configuration on the switch is shown below.

```
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#monitor session 1 source direction rx
ZXR10(config)#interface gei_1/2
ZXR10(config-if)#monitor session 1 source
ZXR10(config)#interface gei_3/3
ZXR10(config-if)#monitor session 1 destination
```

To monitor the data received by gei_1/1, gei_1/2 and gei_2/2, the configuration on the switch can be configured either in interface configuration mode or global configuration mode. Configuration in global configuration mode is shown below.

```
ZXR10(config)#monitor session 1 source gei_1/1-2,gei_2/2
direction rx destination gei_3/3
```

Port mirroring parameters can be deleted either one by one in interface configuration or batch in global configuration mode. Configuration to delete the source port parameters of session 1 is shown below.

```
ZXR10(config)#no monitor session 1 source gei_1/1-2,gei_2/2
```

**Note:**

In global configuration, the values of data flow direction on the source ports are set to the same.

Configuration information of port mirroring is shown below.

```
ZXR10(config)#show monitor session 1
Session 1
-----------------------------------------------
Source Ports:
 Port: gei_1/1            Monitor Direction: rx
 Port: gei_1/2            Monitor Direction: both
Destination Port:
 Port: gei_3/3
-----------------------------------------------
```

# ERSPAN Configuration

## ERSPAN Overview

Port mirroring can be divided into SPAN, RSPAN and ERSPAN:

- SPAN indicates copying packets on one or more ports (source port) to a monitoring port (destination port) of this device for packet monitoring and analysis. Here source port and destination port must be on one device.
- As for RSPAN, source port and destination port are unnecessary to be on one device and they can cross multiple network devices. At present, RSPAN function can pass through L2 network but fails to pass through L3 network. Source port device supports port mirroring or VLAN mirroring.
- As for RSPAN, source port and destination port are unnecessary to be on one device and they can cross multiple network devices. What's more, it can pass through L3 network and is an ideal remote mirroring mode. Source port device supports port mirroring or VLAN mirroring.

**FIGURE 16 ERSPAN EXAMPLE**

ERSPAN implements the following functions: mirroring of original traffic and GRE encapsulation on source-port device, common IP packet forwarding on intermediate device, and mirroring on destination-port device. Function implementation on intermediate device is not illustrated here.

- Source device: Oirt traffic or vlan traffic can be used as source traffic of mirroring; mirrored traffic is sent to intermediate device through designated port after GRE encapsulation.

  Specify source port or mirroring source on source device: Configure soure IP and destination IP of GRE tunnel; configure ERSPAN ID for this mirroring. Additionally, TTL, ip pre/dscp of mirrored packet and VRF ID can be specified.

- Destination device: De-encapsulate mirrored GRE-encapsulated packets received on designated port and send them to test device through designated mirror destination port.

  Specify mirror destination port on destination device; configure destination IP of GRE tunnel; specify corresponding ERSPAN ID for this mirroring.

# Configuring ERSPAN

## Establishing One ERSPAN Session

| Command | Functions |
|---|---|
| ZXR10(config)#**monitor session** <*session-number*> | This establishes one ERSPAN session. |

## Adding Source or Destination Port to Session Entry

| Step | Command | Functions |
|---|---|---|
| 1 | ZXR10(config)#**interface** < *interface-name*> | Enter interface configuration mode. |
| 2 | ZXR10(config-if)#**monitor session** <*session-number*>{**source**{[**direction** {**both**\|**tx**\|**rx**\|**cpu-rx**\|**cpu-tx**\|**cpu-both** }]}\|**destination erspanflags**{**enable**\|**disable**}**tpid 0x8100 ttl**<*ttl_number*> **128 vlan-id** <*vlan-id*>} | This adds source or destination port to session entry. |

## Displaying Session Details Configured by User

| Command | Functions |
|---|---|
| ZXR10(config)#**show monitor session** {**all** \|<*session-number*>} | This displays session details configured by user. |

# ERSPAN Configuration Example

**FIGURE 17 ERSPAN CONFIGURATION EXAMPLE**

As shown in **Figure 1**, set up a tunnel between Switch1 and Switch2, use interface gei_1/1 of Switch1 as mirror source port, and configure ERSPAN mirroring. With this configuration, packets passing through interface gei_1/1 of Switch1 will be encapsulated

with ERSPAN head and mirrored to interface gei_1/1 of Switch2. Configurations are as follows:

Configuration of Switch1:

```
ZXR10(config)#interface gei_1/1 ZXR10(config-gei_1/1)#monitor session 1 source directio
```

Configuration of Switch2:

```
ZXR10(config-gei_1/1)#switchport access vlan 3 ZXR10(config-gei_1/1)#exit ZXR10(config)
```

# Port Loop Detection Configuration

## Port Loop Detection Overview

With port loop detection function, the switch can detect whether there is a loop on the port. If there is a loop, the switch will take measures. This can avoid broadcast storm.

On ZXR10 8900 series switch, port loop detection function can be configured to detect loop on a port or all ports. By default, the detection function is disabled. The switch supports detection function based on VLAN, that is, the switch can detect loop in the VLAN that owns the same PVID with that on the port, as well as in the VLAN that users designate. On a port, it is up to detect loops in 8 VLANs at the same time.

A port sends a Layer 2 multicast message every 15 seconds. If there is a loop on a port, the multicast message will go back to the port through which the message is sent.

## Configuring Port Loop Detection

To configure port loop detection function, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**loop-detect interface** <*port_name*>{**enable** \| **disable**} | This configures port loop detection function on one port or multiple ports |
| 2 | ZXR10(config)#**loop-detect interface** <*port_name*> **vlan** <*vlan_id*>{**enable** \| **disable**} | This configures port loop detection function in a VLAN or multiple VLANs that a port belongs to |
| 3 | ZXR10(config)#**loop-detect portstate** {**block**\| **normal** \| **protect**}<*port_name*> | This configures the state of loop port |

| Step | Command | Function |
|---|---|---|
| 4 | ZXR10(config)#**loop-detect reopen-time** *<1-16777216>* | This configures the reopen time of loop port |
| 5 | ZXR10#**show loop-detect interface** [*<port-name>*] | This views information on a port that enables loop detection function |
| 6 | ZXR10#**show loop-detect reopen-time** | This views reopen time |

**Note:**

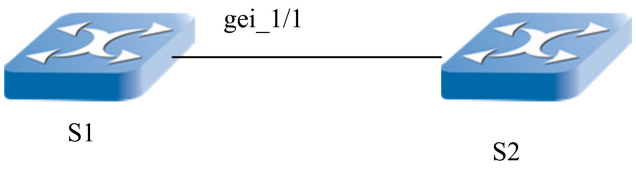
- In the command of step 1, the value of the parameter <port_name> can be a port or multiple port, such as gei_1/1 and gei_1/1-4.
- In the command of step 2, The value of the parameter <vlan_id> can be a VLAN or multiple VLANs, such as vlan 1 and vlan 1-4.
- In the command of step 3, When the switch detects that there is a loop on a port, the switch takes measures according to corresponding configuration.
  - If the configuration is block, the data flow breaks off. The state of the port does not turn down. System generates an alarm.
  - If the configuration is normal, the data flow breaks off, and the state of the port turns down. System generates an alarm.
  - If the configuration is protect, the data flow does not break off. The state of the port does not turn down. System generates an alarm.
  - By default, the configuration is normal.
- In the command of step 4, by default, the time is 10 minutes.

# Port Loop Detection Configuration Example

This example shows how to configure loop detection function.

As shown in Figure 18, gei_1/1 on S1 belongs to VLAN1 and VLAN2. Port loop detection function is enabled on gei_1/1 in VLAN1 and VLAN2.

**FIGURE 18 PORT LOOP DETECTION CONFIGURATION EXAMPLE**




Configuration on S1:

```
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#switchport mode trunk
ZXR10(config-if)#switchport trunk vlan 1-2
ZXR10(config-if)#exit
ZXR10(config)#loop-detect interface gei_1/1 enable
ZXR10(config)#loop-detect interface gei_1/1 vlan 1-2 enable
ZXR10(config)#loop-detect reopen-time 5
```

The information on gei_1/1 is shown below.

```
ZXR10#show loop-detect interface gei_1/4
Interface  Monitor  State      VlanRange
-----------------------------------------------------
gei_1/4    YES       normal      1-2
```

The reopen-time on gei_1/1 is shown below.

```
ZXR10#show loop-detect reopen-time
The reopen time of loop detect  :    5(minute)
```

Chapter 6

# Network Protocol Configuration

**Table of Contents**

IP Address Configuration .....59
ARP Configuration.....61

# IP Address Configuration

## IP Address Overview

IP address is network layer address in the IP protocol stack. One IP address is composed of two parts:

- Network bit identifying the network to which this IP address belongs.
- Host bit identifying a certain host in the network.

**Address Classification**

IP addresses are divided into five classes: A, B, C, D and E. Front three classes are commonly used. Addresses of class D are network multicast addresses and addresses of class E are reserved classes. Range of each class is shown in Table 5.

**TABLE 5 IP ADDRESS FOR EACH CLASS**

| Class | Prefix Characteristic Bit | Network Bit | Host Bit | Range |
|---|---|---|---|---|
| Class A | 0 | 8 | 24 | 0.0.0.0 to 127.255.255.255 |
| Class B | 10 | 16 | 16 | 128.0.0.0 to 191.255.255.255 |
| Class C | 110 | 24 | 8 | 192.0.0.0 to 223.255.255.255 |

| Class | Prefix Characteristic Bit | Network Bit | Host Bit | Range |
|---|---|---|---|---|
| Class D | 1110 | Multicast address | | 224.0.0.0 to 239.255.255.255 |
| Class E | 1111 | Reserved | | 240.0.0.0 to 255.255.255.255 |

Some addresses of Class A, B and C are reserved for private networks. It is recommended that the internal network should use the private network address. They are:

- Class A: 10.0.0.0 to 10.255.255.255
- Class B: 172.16.0.0 to 172.31.255.255
- Class C: 192.168.0.0 to 192.168.255.255

This address classification method is to facilitate routing protocol designing. From this method it can be known the network type just by the prefix characteristic bit of the IP address. This method, however, cannot make the best of the address space. With the dramatic expansion of Internet, problem of address shortage becomes increasingly serious.

**Network, Subnet and Host Bit**

To make most of IP addresses, network can be divided into multiple subnets. Borrow some bits from the highest bit of the host bit as the subnet bit. Remaining part of the host bit still serves as the host bit. IP address is composed of three parts: network bit, subnet bit and host bit.

Network bit and subnet bit identify a network uniquely. Subnet mask is used to decide which parts of IP address are the network bits, subnet bit and host bit. The part with the subnet mask being 1 corresponds to the network bit and subnet bit of the IP address. Part with the subnet mask being 0 corresponds to the host bit.

Division of the subnet greatly improves the utilization of the IP address, and alleviates the problem of IP address shortage.

Some conventions for IP addresses:

- 0.0.0.0 is used when the host without an IP address is started. Address is obtained through RARP, BOOTP and DHCP. This address is also used as a default route in the routing table.
- 255.255.255.255 is used for the destination address of broadcast and cannot be used as a source address.
- 127.X.X.X is called loop-back address. When the actual IP address of the host is not known, this address is used to represent "this host".
- Address with only the host bit being 0 indicates the network itself. Address with the host bit being 1 is the broadcast address of the network.
- Network part or the host part of a valid host IP address cannot be all 0 or 1.

## Configuring IP Address

To configure IP address, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** <*interface -name*> | This enters interface configuration mode |
| 2 | ZXR10(config-if)#**ip address** <*ip-address*><*net-mask*>[<*broadcast-address*>][**secondary**] | This sets interface IP address |
| 3 | ZXR10(config)#**show ip interface** | This views interface IP address |

## IP Address Configuration Example

Assuming that Layer 3 interface VLAN1 is created in ZXR10 8900 series switch, configure the IP address of the interface to 192.168.3.1, and mask to be 255.255.255.0. The configuration is shown below.

```
ZXR10(config)#interface vlan 1
ZXR10(config-if)#ip address 192.168.3.1 255.255.255.0
```

# ARP Configuration

## ARP Overview

A network device should know the IP address of the destination device and its physical address (MAC address) when transmitting data to another network device. The function of Address Resolution Protocol (ARP) is mapping IP address to physical address to ensure successful communication.

First, the source device broadcast carries the ARP request of destination device IP address, so all devices in the network will receive this ARP request. If a device finds that the IP address in the request and its own IP address match, it will transmit a response containing MAC address to source device. The source device obtains the MAC address of the current device through this response.

The mapping relationship between IP address and MAC address is cached in the local ARP table with the purpose of reducing ARP packets in the network to transmit data more rapid. When the device needs to transmit data, it will search ARP table according to IP address, if MAC address of destination device is found in the ARP table, transmitting ARP request is not needed. Dynamic

entries in the ARP table will be deleted automatically after a period of time, which is called ARP aging time.

# Configuring ARP

To configure ARP, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config-if)#**arp timeout** <*seconds*> | This configures aging time of ARP entries on a Layer 3 interface |
| 2 | ZXR10#**clear arp-cache** [**permanent** \| **static** \|{**interface** <*interface-name*>}] | This clears dynamic ARP entries |
| 3 | ZXR10(config)#**arp protect**{ **interface** \| **mac**\| **whole** } **limit-num** <*limit number*> | This configures ARP protection information |
| 4 | ZXR10(config)#**arp to-static** | This turns dynamic ARP to static ARP |
| 5 | ZXR10(config-if)#**set arp** {**permanent** \| **static**}<*ip-address*><*mac-address*> | This configures ARP binding on a Layer 3 interface |
| 6 | ZXR10(config)#**ip arp inspection vlan** <*vlan-id*> | This configures dynamic ARP inspection on a Layer 3 interface |
| 7 | ZXR10(config-if)#**arp learn** | This enables ARP learning on a Layer 3 interface |
| 8 | ZXR10(config-if)#**arp source-filtered** | This configures ARP source filtration on a Layer 3 interface |
| 9 | ZXR10(config-if)#**ip proxy-arp** | This configures ARP proxy on a Layer 3 interface |

# ARP Configuration Example

This example shows how to configure ARP.

```
ZXR10(config)#interface vlan 1
ZXR10(config-if)#arp timeout 1200
```

To view ARP entries of specified interface, use the following command.

| Command | Function |
|---|---|
| ZXR10**show arp** [**interface**<*interface-name*>] | This views ARP entries of specified interface |

**Example** This example shows how to view ARP table of Layer 3 interface VLAN1.

```
ZXR10#show arp interface vlan 1
Address    Age(min)    Hardware Addr    Interface
10.1.1.1     -         000a.010c.e2c6      vlan1
10.1.100.100 18      00b0.d08f.820a       vlan1
ZXR10#
```

To view ARP entries with keepalive attribute, use the following command.

| Command | Function |
|---|---|
| ZXR10**show arp-rt** | This views ARP entries with keepalive attribute |

# ARP Query Example

To view ARP entry with designated external VLAN-ID and internal VLAN-ID, use the following command.

| Command | Function |
|---|---|
| ZXR10#**show arp** [**exvlanID** <*id*>][**invlanID** <*id*>] | This views ARP entry with designated external VLAN-ID and internal VLAN-ID |

**Example** This example shows how to view ARP table with external VLAN-ID of 21 and internal VLAN-ID of 31.

```
ZXR10#show arp exvlanID 21 invlanID 31
Arp protect whole is disabled
The count is 2
IPAddress Age HardwareAddress interface ExVlanID InVlanID
-----------------------------------------------------------
10.1.1.1   S   0000.0000.0001   qinq1     21         31
10.1.1.2   S   0000.0000.0001   qinq1     21         31
```

This page is intentionally blank.

Chapter 7

# DHCP Configuration

**Table of Contents**

DHCP Overview ..... 65
DHCP Snooping Overview ..... 66
Configuring DHCP ..... 66
DHCP Configuration Examples ..... 68
DHCP Maintenance and Diagnosis ..... 71

## DHCP Overview

DHCP allows a host on a network to obtain an IP address for normal communications and related configuration information from a DHCP server. Details of DHCP are described in RFC 2131.

**Working Procedure**

DHCP uses UDP as the transmission protocol. The host sends messages to port 67 of the DHCP server, who will return messages to port 68 of the host. A DHCP works in the following steps:

1. A host sends a DHCP Discover broadcast message requesting an IP address and other configuration parameters.
2. A DHCP server returns a DHCP Offer message containing a valid IP address.
3. Host selects the server at which the DHCP Offer arrives first, and sends a DHCP Request message to the server, which indicates it accepts the related configurations.
4. Selected DHCP server returns a DHCP Ack message for acknowledgement.

By now the host can use the IP address and relevant configuration obtained from the DHCP server for communication.

DHCP supports three mechanisms for IP address allocation:

- DHCP assigns a permanent IP address to a client.
- DHCP assigns an IP address to a client for a limited period of time (or until the client explicitly relinquishes the address).
- Network administrator assigns an IP address to a client and DHCP is used simply to convey the assigned address to the client.

Usually Dynamic allocation method is adopted. The valid time segment of using the address is called lease period. Once the lease period expires, the host must request the server for continuous lease. The host cannot continue to lease until the server accepts the request, otherwise it must give up unconditionally.

**DHCP Relay** Routers do not send the received broadcast packet from a sub-network to another by default. But the router as the default gateway of the client host must send the broadcast packet to the sub-network where the DHCP server locates when the DHCP server and client host are not in the same sub-network. This function is called DHCP relay.

ZXR10 8900 series switch can act as a DHCP server or DHCP relay to forward DHCP information.

# DHCP Snooping Overview

DHCP brings convenience for IP address allocation, but it also brings problems.

DHCP service allows multiple DHCP servers to exit in a subnet. Therefore, the administrator cannot ensure that IP addresses of users are allocated by the designated DHCP server. The addresses may be allocated by DHCP servers that are set by other users illegally.

In a DHCP service subnet, hosts with legal IP addresses and masks can access this subnet. DHCP server may allocate these legal addresses to other hosts. This causes address confliction.

To solve the above problems, ZXR10 8900 series switch uses DHCP snooping function to prevent bogus DHCP server in a subnet. The port connecting with DHCP server must be set as trust port. Combining with dynamic ARP inspection technology, DHCP snooping function prevents binding of illegal IP and MAC. This ensures the server to allocate IP addresses correctly.

# Configuring DHCP

## Configuring DHCP Server

To configure DHCP server, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**ip dhcp enable** | This enables DHCP server process globally. |
| 2 | `ZXR10(config)#`**ip local pool** <*pool-name*><*low-ip-address*><*high-ip-address*><*net-mask*> | This configures an IP address pool for a DHCP server. |
| 3 | `ZXR10(config)#`**ip dhcp server leasetime** <*time*> | This sets the lease time of the IP address leased by a DHCP server to client. |

| Step | Command | Function |
|---|---|---|
| 4 | ZXR10(config)#**ip dhcp server dns** <*mdns-address*>[<*sdns-address*>] | This sets DNS address advertised by a DHCP server to client. |
| 5 | ZXR10(config)#**interface vlan**<*vlan-number*> | This accesses VLAN L3 interface. |
| 6 | ZXR10(config-if)#**ip dhcp mode server** | This enables DHCP on an interface. |
| 7 | ZXR10(config-if)#**ip dhcp server gateway** <*ip-address*> | This configures default gateway address for one client. |
| 8 | ZXR10(config-if)#**peer default ip pool** <*pool-name*> | This applies defined IP address pool on L3 interface. |

# Configuring DHCP Relay

To configure DHCP relay, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**ip dhcp enable** | This enables DHCP process |
| 2 | ZXR10(config)#**interface vlan**<*vlan-number*> | This enters Layer 3 VLAN interface configuration mode |
| 3 | ZXR10(config-if)#**ip dhcp mode relay** | This configures DHCP relay on an interface |
| 4 | ZXR10(config-if)#**ip dhcp relay server** <*ip-address*> **ip dhcp relay agent** <*ip-address*> | This configures DHCP relay agent |
| 5 | ZXR10(config-if)#**ip dhcp relay server** <*ip-address*>{**security** \| **standard**} | This configures IP address of external DHCP server |

**Note:**

In the command of Step 5, when the mode is set to **security**, the address of DHCP server displayed on DHCP Client is the address of relay agent. When the mode is set to **standar**d, the address of DHCP server displayed on DHCP Client is actually the address of the server. Therefore, the **security** mode can protect the server from attack.

# Configuring DHCP Snooping

To configure DHCP snooping, perform the following steps.

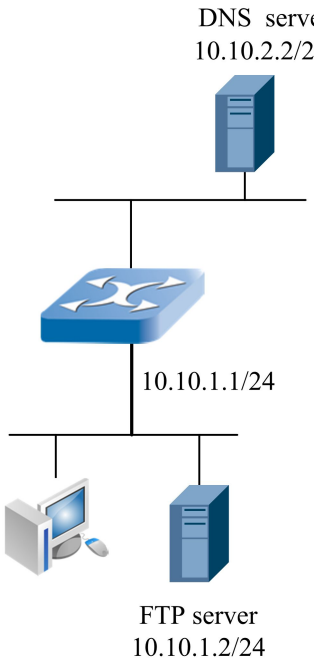
| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**ip dhcp snooping enable** | This enables DHCP snooping process |
| 2 | ZXR10(config)#**ip dhcp snooping vlan** <*vlan-id*> | This enables DHCP snooping in a VALN |
| 3 | ZXR10(config)#**ip dhcp snooping trust** <*port-number*> | This configures an interface on DHCP server to be a trust interface |
| 4 | ZXR10(config)#**ip dhcp snooping binding** <*mac-address*> **vlan** <*vlan-id*><*ip-address*><*port-number*> **expiry** <*time*> | This adds an entry to DHCP Snooping database |
| 5 | ZXR10(config)#**ip arp inspection vlan** <*vlan-id*> | This configures dynamic ARP inspection |

# DHCP Configuration Examples

## DHCP Server Configuration Example

The switch acts as the DHCP server and default gateway. The host obtains IP address through the DHCP dynamically, as shown in Figure 19.

**FIGURE 19 DHCP SERVER CONFIGURATION EXAMPLE**



Configuration on the switch:

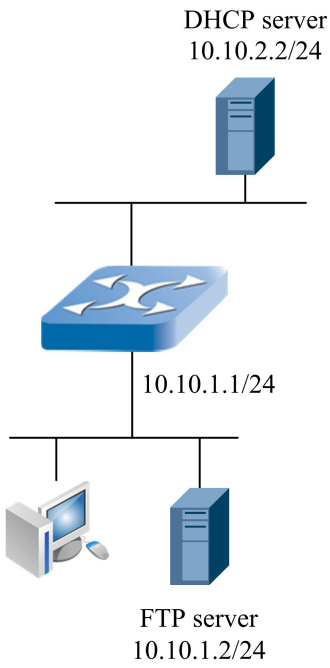
```
ZXR10(config)#ip dhcp server dns 10.10.2.2
ZXR10(config)#ip dhcp server leasetime 90
ZXR10(config)#ip local pool dhcp 10.10.1.3 10.10.1.254 255.255.255.0
ZXR10(config)#interface vlan10
ZXR10(config-if)#ip dhcp mode server
ZXR10(config-if)#ip address 10.10.1.1 255.255.255.0
ZXR10(config-if)#ip dhcp server gateway 10.10.1.1
ZXR10(config-if)#peer default ip pool dhcp
ZXR10(config-if)#exit
ZXR10(config)#ip dhcp enable
```

# DHCP Relay Configuration Example

When DHCP client and server are not in the same sub-network, the router which connects with users works as a DHCP relay.

The switch enables DHCP relay function and a single server 10.10.2.2 provides DHCP server function. This mode is usually adopted when a lot of hosts require the DHCP service. This is shown in Figure 20.

**FIGURE 20 DHCP RELAY CONFIGURATION EXAMPLE**




Configuration on the switch:

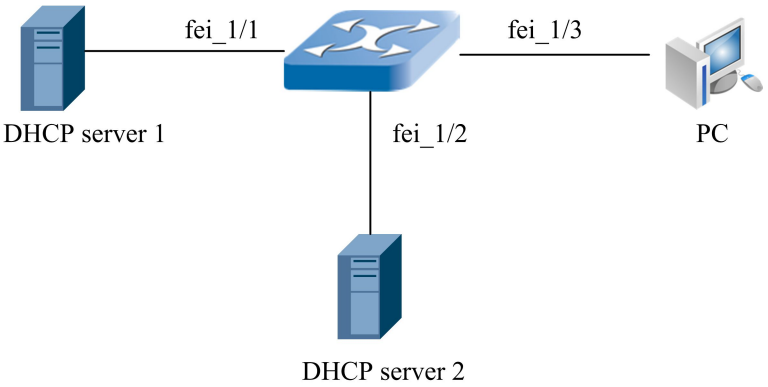
```
ZXR10(config)#interface vlan10
ZXR10(config-if)#ip dhcp mode relay
ZXR10(config-if)#ip address 10.10.1.1 255.255.255.0
ZXR10(config-if)#ip dhcp relay agent 10.10.1.1
ZXR10(config-if)#ip dhcp relay server 10.10.2.2 security
ZXR10(config-if)#exit
ZXR10(config)#ip dhcp enable
```

# DHCP Snooping Preventing False DHCP Server Configuration Example

DHCP server 1 connects with fei_1/1 of the switch. DHCP Server 1 is configured by administrator. DHCP server 2 connects with fei_1/2 of switch, and it is a private and illegal server. Fei_1/1 and fei_1/2 belong to vlan100. Enable DHCP snooping function on the switch to prevent setting false DHCP server in the network, as shown in Figure 21.

At this time, it is required to enable DHCP snooping function in vlan100 and set fei_1/1 as a trust port.

**FIGURE 21 DHCP SNOOPING PREVENTING FALSE DHCP SERVER**



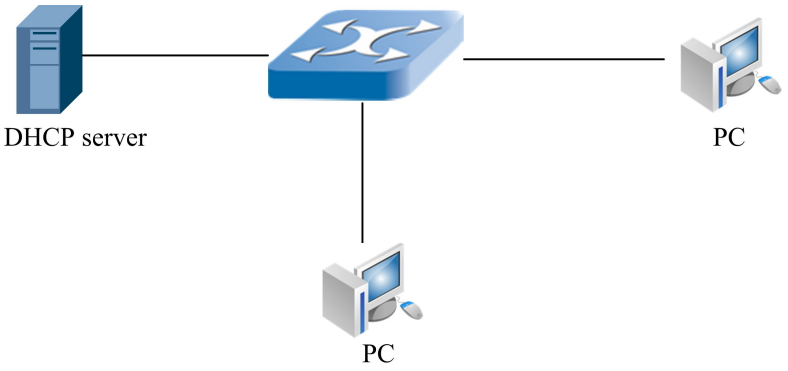

Configuration on the switch:

```
ZXR10(config)#interface fei_1/1
ZXR10(config-if)#sw ac vlan 100
ZXR10(config)#interface fei_1/2
ZXR10(config-if)#sw ac vlan 100
ZXR10(config)#vlan 100
ZXR10(config-vlan)#ip dhcp snooping
ZXR10(config)#ip dhcp snooping enable
ZXR10(config)#ip dhcp snooping vlan 100
ZXR10(config)#ip dhcp snooping trust fei_1/1
```

# DHCP Snooping Preventing Static IP Configuration Example

DHCP server belongs to vlan100 and the PCs belong to vlan200. The PC gets IP address through the server. At this time it is necessary to forbid the PCs to set static IP address through DHCP snooping and dynamic ARP inspection. This is shown in Figure 22.

**FIGURE 22 DHCP SNOOPING PREVENTING STATIC IP**




Configuration on the switch:

```
ZXR10(config)#ip dhcp snooping enable
ZXR10(config)#ip dhcp snooping vlan 100
ZXR10(config)#ip arp inspection vlan 100
```

# DHCP Maintenance and Diagnosis

To configure DHCP maintenance and diagnosis, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show ip dhcp server user slot** <*slot-id*> | This displays list of current online users on DHCP server process module |
| 2 | ZXR10#**show ip local pool** [<*pool-name*>] | This displays configuration information of local address pools |
| 3 | ZXR10#**show ip interface** | This displays configuration information of DHCP server/relay related to an interface |
| 4 | ZXR10#**show ip dhcp snooping configure** | This displays DHPC snooping global configuration information |
| 5 | ZXR10#**show ip dhcp snooping vlan** [<*vlan-id*>] | This displays configuration information of VLAN that enables DHCP snooping function |
| 6 | ZXR10#**show ip dhcp snooping trust** | This displays configuration information of DHCP snooping trust interface |

| Step | Command | Function |
|---|---|---|
| 7 | ZXR10#**show ip dhcp snooping database slot** <*slot-id*> | This views information in DHCP Snooping database |
| 8 | ZXR10#**show ip arp inspection vlan** [<*vlanl-id*>] | This displays configuration information of VLAN that enables dynamic ARP inspection function |
| 9 | ZXR10#**debug ip dhcp** | This tracks packet sending and receiving as well as processing on DHCP server/relay |

Chapter 8

# VRRP Configuration

**Table of Contents**

VRRP Overview .....73
Configuring VRRP .....74
VRRP Configuration Examples .....74
VRRP Maintenance and Diagnosis.....76

# VRRP Overview

Host in a broadcast domain usually sets a default gateway as the next hop of routing data packets. The host in the broadcast domain cannot communicate with the host in another network unless the default gateway works normally. To avoid the single point of failure caused by the default gateway, multiple router interfaces are configured in the broadcast domain and run the Virtual Router Redundancy Protocol (VRRP) in these routers.

VRRP is used to configure multiple router interfaces in a broadcast domain into a group to form a virtual router and assigns an IP address to the router to function as its interface address. This interface address may be the address of one of router interfaces or the third party address.

If the interface address is used, a router with the interface address acts as the master router. Other routers act as the backup routers. The router with high priority is used as the master router if the third party address is used. If two routers have the same priority, the one that sends VRRP message first wins.

Set the IP address of the virtual router to gateway on the host in this broadcast domain. The master router is replaced with the backup router with the highest priority if the master router is faulty, without affecting the host in this domain. The host in this domain cannot communicate with outside world only when all routers in the VRRP group work abnormally.

These routers can be configured into multiple groups for mutual backup. The hosts in the domain use different IP addresses as gateway to implement data load balance.

# Configuring VRRP

To configure VRRP, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface vlan**<*vlan-number*> | This enters Later 3 VLAN interface configuration mode |
| 2 | ZXR10(config-if)#**vrrp** <*group*> **ip** <*ip-address*>[**secondary**] | This sets a VRRP virtual IP address and runs VRRP on an interface |
| 3 | ZXR10(config-if)#**vrrp** <*group*> **priority** <*priority*> | This configures a VRRP priority, with 100 by default |
| 4 | ZXR10(config-if)#**vrrp** <*group*> **preempt** [**delay** <*seconds*>] | This configures whether to enable preempt |
| 5 | ZXR10(config-if)#**vrrp** <*group*> **advertise** [**msec**]<*interval*> | This configures time interval for sending VRRP advertisements |
| 6 | ZXR10(config-if)#**vrrp** <*group*> **learn** | This learns the time interval from primary gateway to send VRRP messages |
| 7 | ZXR10(config-if)#**vrrp** <*group*> **authentication** <*string*> | This configures authentication character string |
| 8 | ZXR10(config-if)#**vrrp** <*group*> **out-interface** <*interface-name*> | This configures the out interface of VRRP messages |

**Note:**

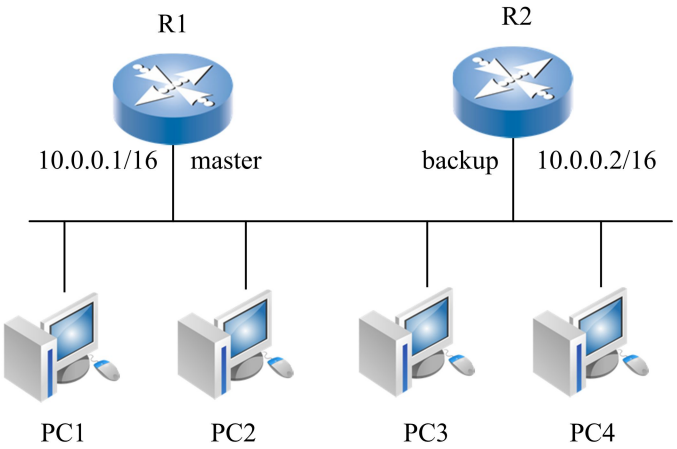
A VRRP group can be configured with multiple virtual addresses. Hosts connected to it can use any one of them as gateway for communications.

# VRRP Configuration Examples

## Basic VRRP Configuration Example

This example shows that R1 and R2 run in the VRRP protocol between each other. R1 interface address 10.0.0.1 is used as the VRRP virtual address, therefore R1 is considered as a master router. This is shown in Figure 23.

**FIGURE 23 BASIC VRRP CONFIGURATION EXAMPLE**




Configuration on R1:

```
ZXR10_R1(config)#interface vlan 1
ZXR10_R1(config-if)#ip address 10.0.0.1 255.255.0.0
ZXR10_R1(config-if)#vrrp 1 ip 10.0.0.1
```

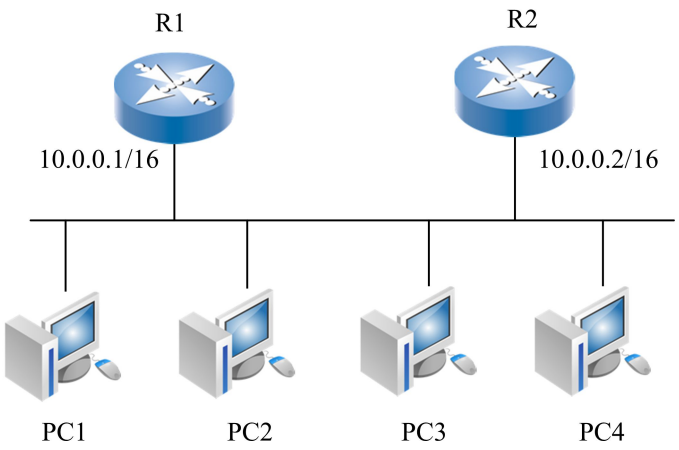
Configuration on R2:

```
ZXR10_R2(config)#interface vlan 1
ZXR10_R2(config-if)#ip address 10.0.0.2 255.255.0.0
ZXR10_R2(config-if)#vrrp 1 ip 10.0.0.1
```

# Symmetric VRRP Configuration Example

Two VRRP groups are booted in this example, where PC1 and PC2 use virtual router in Group 1 as default gateway with address 10.0.0.1. PC3 and PC4 use virtual router in Group 2 as default gateway with address 10.0.0.2. R1 and R2 serve as mutual backup. Four hosts cannot communicate with outside world until both routers become invalid. This is shown in Figure 24.

**FIGURE 24 SYMMETRIC VRRP CONFIGURATION EXAMPLE**




Configuration on R1:

```
ZXR10_R1(config)#interface vlan 1
ZXR10_R1(config-if)#ip address 10.0.0.1 255.255.0.0
ZXR10_R1(config-if)#vrrp 1 ip 10.0.0.1
ZXR10_R1(config-if)#vrrp 2 ip 10.0.0.2
```

Configuration on R2:

```
ZXR10_R2(config)#interface vlan 1
ZXR10_R2(config-if)#ip address 10.0.0.2 255.255.0.0
ZXR10_R2(config-if)#vrrp 1 ip 10.0.0.1
ZXR10_R2(config-if)#vrrp 2 ip 10.0.0.2
```

# VRRP Maintenance and Diagnosis

To configure maintenance and diagnosis, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show vrrp** [<*group*>\|**brief**\|**interface** <*interface-name*>] | This displays configuration information of all VRRP groups |
| 2 | ZXR10#**debug vrrp** {**state**\|**packet**\|**event**\|**error**\|**all**} | This enables the switch for displaying VRRP debugging information |

Chapter 9

# ACL Configuration

**Table of Contents**

ACL Overview ..... 77
NP-Based ACL Overview ..... 78
Configuring ACLs ..... 79
Configuring Event Linkage ACL Rule ..... 85
Applying NP-Based ACL ..... 87
ACL Configuration Example ..... 88
ACL Maintenance and Diagnosis ..... 89

## ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACL can filter traffic as it passes through a router and permit or deny packets at specified interfaces.

An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACL to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. It tests packets against the conditions in an access list one by one. The first match determines whether the switch accepts or rejects the packets because the switch stops testing conditions after the first match. The order of conditions in the list is critical. When there are no conditions matched, the switch rejects the packets. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet.

Packet matching rules defined by the ACL are also used in other conditions where distinguishing traffic is needed. For instance, the matching rules can define the traffic classification rule in the QoS.

ZXR10 8900 series switch provides seven types of ACLs:

- Standard ACL

  Only source IP addresses are matched against the ACL.

- Extended ACL

  Source/destination IP address, IP protocol type, TCP source/destination port number, TCP-control, UDP source/destination port number, ICMP type, ICMP code, DiffServ Code Point (DSCP), ToS and precedence are matched against the ACL.

- Layer 2 ACL

  Source/destination MAC address, source VLAN ID, Layer 2 Ethernet protocol type and 802.1p priority value are matched against the ACL.

- Hybrid ACL

  Source/destination MAC address, source VLAN ID, source/destination IP address, TCP source/destination port number, UDP source/destination port number are matched against the ACL.

- Standard IPv6 ACL

  Only source IPv6 address is matched.

- Extended IPv6 ACL

  Source/Destination IPv6 address is matched.

- User-Defined ACL

  The number of tags and byte offset value are matched.

Each ACL has an access list number to identify. The access list number is a number. The access list number ranges of different types of ACLs are shown in Table 6.

**TABLE 6 ACL DESCRIPTIONS**

| ACL Type | Access List Number |
|---|---|
| Standard ACL | The range is from 1 to 99. The expanded range is from 1000 to 1499. |
| Extended ACL | The range is from 100 to 199. The expanded range is from 1500 to 1999. |
| Layer 2 ACL | The range is from 200 to 299. |
| Hybrid ACL | The range is from 300 to 349. |
| Standard IPv6 ACL | The range is from 2000 to 2499. |
| Extended IPv6 ACL | The range is from 2500 to 2999. |
| User-Defined ACL | The range is from 3000 to 3499. |

Each ACL supports up to 1000 rules with the codes ranging from 1 to 1000.

# NP-Based ACL Overview

To apply the configured ACL to physical port, VLAN or Smartgroup virtual interface, user can choose common processing mode or Network Processor (NP) mode. As for NP processing mode—based ACL, the switch must be configured with NP fastener subcard, or ACL will not be valid.

NP processing mode—based ACL is not conflicted with common processing mode—based ACL. That is, the same object (a physi-

cal port, VLAN or Smartgroup virtual interface) supports two ACL processing modes and can process packets in these two modes.

# Configuring ACLs

ACL configuration includes:

- Define an ACL rule
- Configure a time range
- Apply the ACL to a port

## Defining ACLs

The following issues are to be taken into account when defining ACL rules.

- When a packet meets multiple rules, first rule will be matched. Rule sequence is very important. Generally, rules in a small range are put in the front and rules in a large range are put in the back.
- Considering network security, system will add an implicit **deny** rule to the end of each ACL automatically for denying all the packets. A **permit** rule for allowing all packets should be defined at the end of each ACL.

### Defining Standard ACL

To configure standard ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**acl standard** {**number** <*acl-number*>\|**name** <*acl-name*>\| **alias** <*alias-name*>}[**match-order** {**auto** \| **config**}] | This enters standard ACL configuration mode |
| 2 | ZXR10(config-std-acl)#**rule** <*rule-no*>{**permit**\|**deny**}{<*source*>[<*source-wildcard*>]\|**any**}[**time-range** <*timerange-name*>] | This defines rules |
| 3 | ZXR10(config-std-acl)#**move** <*rule-no*> **after** <*rule-no*> | This moves a rule |
| 4 | ZXR10(config-std-acl)#**attach time-range** <*Time range name*> **to** <*rule id*> | This binds a time range to a rule |

**Example** This example describes how to define a standard ACL which allows access of messages from network 192.168.1.0/24 but denies messages from source IP address 192.168.1.100.

```
ZXR10(config)#acl basic number 10
ZXR10(config-std-acl)#rule 1 deny 192.168.1.100 0.0.0.0
```

```
ZXR10(config-std-acl)#rule 2 permit 192.168.1.0 0.0.0.255
```

## Defining Extended ACL

To configure extended ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**acl extend** {**number** *<acl-number>*\|**name** *<acl-name>*\| **alias** *<alias-name>*}[**match-order** {**auto**\|**config**}] | This enters extended ACL configuration mode |
| 2 | ZXR10(config-ext-acl)#**rule** *<rule-no>*{**permit**\|**deny**} **icmp** {*<source><source-wildcard>*\|**any**}{*<dest><dest-wildcard>*\|**any**}[*<icmp-type>*[**icmp-code** *<icmp-code>*]][**precedence** *<pre-value>*][**tos** *<tos-value>*][**dscp** *<dscp-value>*][**time-range** *<timerange-name>*] | This defines ICMP-based rules |
| | ZXR10(config-ext-acl)#**rule** *<rule-no>*{**permit**\|**deny**}{*<ip-number>*\|**ip**}{*<source><source-wildcard>*\|**any**}{*<dest><dest-wildcard>*\|**any**}[{[**precedence** *<pre-value>*][**tos** *<tos-value>*]}\|**dscp** *<dscp-value>*][**time-range** *<timerange-name>*] | This defines rules on the basis of IP or IP protocol code |
| | ZXR10(config-ext-acl)#**rule** *<rule-no>*{**permit**\|**deny**} **tcp** {*<source><source-wildcard>*\|**any**}[*<rule><port>*]{*<dest><dest-wildcard>*\|**any**}[*<rule><port>*][**established**][{[**precedence** *<pre-value>*][**tos** *<tos-value>*]}\|**dscp** *<dscp-value>*][**tcp-control** *<tcp-control-value>*][**time-range** *<timerange-name>*] | This defines TCP-based rules |
| | ZXR10(config-ext-acl)#**rule** *<rule-no>*{**permit**\|**deny**} **udp** {*<source><source-wildcard>*\|**any**}[*<rule><port>*]{*<dest><dest-wildcard>*\|**any**}[*<rule><port>*][{[**precedence** *<pre-value>*][**tos** *<tos-value>*]}\|**dscp** *<dscp-value>*][**time-range** *<timerange-name>*] | This defines UDP-based rules |
| 3 | ZXR10(config-ext-acl)#**move** *<rule-no>* **after** *<rule-no>* | This moves a rule |
| 4 | ZXR10(config-ext-acl)#**attach time-range** *<Time range name>* **to** *<rule id>* | This binds a time range to a rule |

**Example** This example describes how to configure an extended ACL. It is required to implement the following functions:

- Permit UDP packets from network segment 210.168.1.0/24, destination IP address is 210.168.2.10, source port is 100 and destination port is 200 to pass.
- Denies BGP messages from network 192.168.2.0/24.
- Denies all ICMP messages.
- Denies all messages with IP protocol code 8.

```
ZXR10(config)#acl extend number 150
ZXR10(config-ext-acl)#rule 1 permit udp 210.168.1.0 0.0.0.255
Eq 100 210.168.2.10 0.0.0.0 eq 200
ZXR10(config-ext-acl)#rule 2 deny tcp 192.168.2.0 0.0.0.255
Eq BGP any
ZXR10(config-ext-acl)#rule 3 deny icmp any any
```

```
ZXR10(config-ext-acl)#rule 4 deny 8 any any
```

## Defining Layer 2 ACL

To configure Layer 2 ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**acl link** {**number** <*acl-number*>\|**name** <*acl-name*>\| **alias** <*alias-name*>}[**match-order** {**auto** \| **config**}] | This enters Layer 2 ACL configuration mode |
| 2 | ZXR10(config-link-acl)#**rule** <*rule-no*>{**permi t**\|**deny**}<*protocol-number*>[**cos** <*cos-vlaue*>\| **incos** <*cos-vlaue*>\|**dinvlan** <*vlan-id*>\|**doutervlan** <*vlan-id*>][**ingress** {[<*source-vlanid*>][<*source-mac*><*source-mac-wildcard*>\|**any**]}][**egress** {<*de st-mac*><*dest-mac-wildcard*>\|**any**}][**time-range** <*timerange-name*>] | This configures rules in an ACL |
| 3 | ZXR10(config-link-acl)#**move** <*rule-no*> **after** <*rule-no*> | This moves a rule |
| 4 | ZXR10(config-link-acl)#**attach time-range** <*Time range name*> **to** <*rule id*> | This binds a time range to a rule |

**Example** This example describes how to define a L2 ACL which allows access of IP packets with source MAC address 00d0.d0c0.5741 and 802.1p code 5.

```
ZXR10(config)#acl link number 200
ZXR10(config-link-acl)#rule 1 permit ip cos 5
ingress 10 00d0.d0c0.5741 0000.0000.0000
ZXR10(config-link-acl)#rule 2 deny 8847
```

## Defining Hybrid ACL

To configure hybrid ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**acl hybrid** {**number** <*acl-number*>\|**n ame** <*acl-name*>\| **alias** <*alias-name*>} | This enters hybrid ACL configuration mode |
| 2 | ZXR10(config-hybd-acl)#**rule** <*rule-no*>{**permit** \|**deny**}<*protocol-numberl*>{{<*source-ip*><*sour ce-ip-wildcard*>}\|**any**}[**eq** <*port-number*>]{{<*d estination-ip*><*dest-ip-wildcard*>}\|**any**}[**eq** <*port-number*>]{<*ethernet-protocol-number*>\| **any** \|**arp** \| **ip**}[**cos** \| **incos** \| **dinvlan** \| **doutervlan** \| **egress** \| **ingress** \| **time-range**] | This defines rule in an ACL |
| 3 | ZXR10(config-hybd-acl)#**move** <*rule-no*> **after** <*rule-no*> | This moves a rule |
| 4 | ZXR10(config-hybd-acl)#**attach time-range** <*Time range name*> **to** <*rule id*> | This binds a time range to a rule |

**Example** This example describes how to configure a hybrid ACL. It is required to implement the following functions:

- Permit access of UDP messages from network 210.168.1.0/24, destination IP address 210.168.2.10, destination MAC address 00d0.d0c0.5741, source port 100 and destination port 200.
- Denies BGP messages from network 192.168.3.0/24.
- Denies messages from MAC address 0100.2563.1425.

```
ZXR10(config)#acl hybrid number 300
ZXR10(config-hybd-acl)#rule 1 permit udp 210.168.1.0 0.0.0.255 Eq
00 210.168.2.10 0.0.0.0 eq 200 Egress 00d0.d0c0.5741 0000.0000.0000
ZXR10(config-hybd-acl)#rule 2 deny tcp 192.168.3.0 .0.0.255
q BGP any
ZXR10(config-hybd-acl)#rule deny any any
ngress 0100.2563.1425 0000.0000.0000
```

## Defining Standard IPv6 ACL

To configure standard IPv6 ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**ipv6 acl standard** {**number** <*acl-number*>\|**name** <*acl-name*>\| **alias** <*alias-name*>}[**match-order** {**auto** \| **config**}] | This enters standard IPv6 ACL configuration mode |
| 2 | ZXR10(config-std-v6acl)#**rule** <*rule-no*>{**permit**\|**deny**}{<*source*>\|**any**}[**time-range** <*timerange-name*>] | This defines ACL rule |
| 3 | ZXR10(config-std-v6acl)#**move** <*rule-no*>{**after** \| **before**}<*rule-no*> | This moves a rule |
| 4 | ZXR10(config-std-v6acl)#**attach time-range** <*Te range name*> **to** <*rule id*> | This binds a time range to a rule |

**Example** This example shows how to configure standard IPv6 ACL. It defines an ACL that allows packets from network segment 3001::/16 to pass.

```
ZXR10(config)#ipv6 acl standard number 2000
ZXR10(config-std-v6acl)#rule  1 permit 3001::/16
```

## Defining Extended IPv6 ACL

To configure extended IPv6 ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**ipv6 acl extended** {**number** <*acl-number*>\|**name** <*acl-name*>\| **alias** <*alias-name*>}[**match-order** {**auto** \| **config**}] | This enters extended IPv6 ACL configuration mode |
| 2 | ZXR10(config-ext-v6acl)#**rule** <*rule-no*>{**permit**\|**deny**} **ip** {<*source*>\|**any**}{<*dest*>\|**any**}[**time-range** <*timerange-name*>] | This defines ACL rule |

| Step | Command | Function |
|---|---|---|
| 3 | ZXR10(config-ext-v6acl)#**move** <*rule-no*>{**after** \| **before**}<*rule-no*> | This moves a rule |
| 4 | ZXR10(config-ext-v6acl)#**attach time-range** <*Time range name*> **to** <*rule id*> | This binds a time range to a rule |

**Example** This example shows how to configure extended IPv6 ACL. It defines an ACL that allows packets from network segment 3000::/16 to 4000::/16 to pass.

```
ZXR10(config)#ipv6 acl extended 2500
ZXR10(config-ext-v6acl)#rule 1 permit 3000::/16 4000::/16
```

## Defining Customized ACL

To configure customized ACL, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**acl user-defined** {**number** <*3000-3499*>\| **name** <*acl-name*>\| **alias** < *alias-name*>} | This enters basic ACL configuration mode |
| 2 | ZXR10(config-user-acl)#**rule** <*rule-id*>{**permit** \| **deny**}{**any** \|{**tag** <*tag-num*><*offset*><*rule-string*><*rule-mask*>**&**<*1-4*>}}[**time-range** < *timerange-name*>] | This defines ACL rule |
| 3 | ZXR10(config-user-acl)#**move** <*rule-no*>{**after** \| **before**}<*rule-no*> | This moves a rule |
| 4 | ZXR10(config-user-acl)#**attach time-range** <*Time range name*> **to** <*rule id*> | This binds a time range to a rule |

**Example** This example shows how to configure extended IPv6 ACL.

A user defines an ACL to allow packets with the following features to pass:

- Tag is 1.
- Rule is 0x1111.
- Mask is 0x000f.
- Offset is 4 bytes.

```
ZXR10(config)#acl user-define number 3000
ZXR10(config-user-acl)#rule 1 permit tag 1 4 0x1111 0x000f
```

# Configuring Time Range

To configure time range, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**time-range enable** | This enables time range function |
| 2 | ZXR10(config)#**time-range** <*time-range-name*> | This enters time range configuration mode |
| 3 | ZXR10(config-tr)#**absolute start** <*hh:mm:ss*><*mm-dd-yyyy*>[**end** <*hh:mm:ss*><*mm-dd-yyyy*>] | This configures absolute time range |
| 4 | ZXR10(config-tr)#**periodic** {**daily** \| **monday** \| **tuesday** \| **wednesday** \| **thursday** \| **friday** \| **staturday** \| **sunday** \| **weekdays** \| **weekend**}<*hh:mm:ss*> **to** {**daily** \| **monday** \| **tuesday** \| **wednesday** \| **thursday** \| **friday** \| **staturday** \| **sunday** \| **weekdays** \| **weekend**}<*hh:mm:ss*> | This configures periodic time range |

**Note:**

Configuration of time range has the following situations:

- Configuration of absolute time range: configure the start time and end time of the time range.
- Configuration of periodic time range: configure the start time and end time of the period.

# Applying ACL to Physical Port

To apply ACL to physical ports, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** <*port-name*> | This enters port configuration mode |
| 2 | ZXR10(config-if)#**ip access-group** <*acl-number*>{**in**\|**out**\|**vfp**} | This binds ACL to physical ports |

**Note:**

Each physical port has "in" and "out" direction. ACL can only be applied on either of the directions. A new configured ACL covers the old ACL.

For example, the following commands are configured in port configuration mode.

```
ZXR10(config-if)#ip access-group 10 in
ZXR10(config-if)#ip access-group 100 in
```

In this situation, only ACL 100 is effective on this port in "in" direction. Configuration in "out" direction is similar.

When the following commands are configured on a port, ACL 10 is effective on this port in "in" direction and ACL 100 is effective on this port in "out" direction.

```
ZXR10(config-if)#ip access-group 10 in
ZXR10(config-if)#ip access-group 100 out
```

## Applying ACL to Virtual Port

To apply ACL to virtual port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**vlan** <*vlan-number*> | This enters VLAN configuration mode |
| 2 | ZXR10(config-vlan)#**ip access-group** <*acl-number*> **in** | This applies ACL to a virtual port |

# Configuring Event Linkage ACL Rule

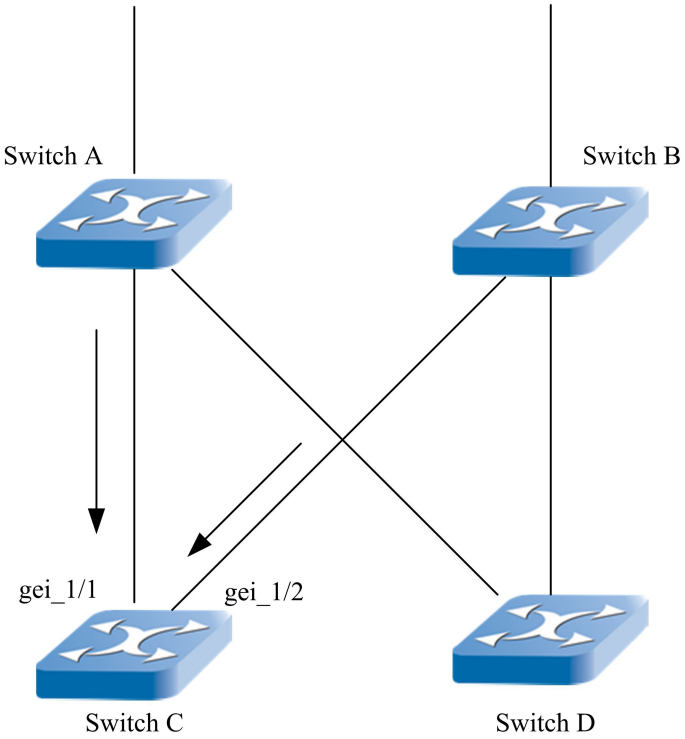
After event linkage ACL rule is configured, when two interfaces on a device are connected to an upper layer device, only enable one interface. If one interface status turns to down, the other interface is enabled automatically.

To configure linkage ACL rule, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**event-list** <*name*> | This creates an event list. |
| 2 | ZXR10(config-event)#**interface** <*interface-name*>{**admin** \| **physical** \| **protocol**}{**down** \| **up**} | This sets the conditions of triggering event, where port management state, physical state and protocol state can be set. |
| 3 | ZXR10(config-event)#**exit** | This exits event list. |
| 4 | ZXR10(config)#**acl standard number** <*number*> | This enters standard access list. |
| 5 | ZXR10(config-std-acl)#**rule 1 permit** <*source-address*><*source-wildcard*> **event** <*name*> | This associates the ACL rule with the event. |

**Example** As shown in Figure 25, Switch A and Switch B back up for each other. Switch C receives two same data flows. To avoid this phenomenon, an event linkage ACL rule is configured.

**FIGURE 25 CONFIGURING EVENT LINKAGE ACL RULE**




How to configure?

1. Define one event list. The prerequisite of event trigger is that interface gei_1/1 is down;
2. Define one standard ACL, where rule 1 permits all packets to pass through, rule 2 denies all packets. By associating rule 1 with event, execute rule 1 when protocol on interface gei_1/1 is down;
3. Apply ACL on "in" direction of interface gei_1/2.

Configuration of Switch C:

```
ZXR10(config)#event-list zte
ZXR10(config-event)#interface gei_1/1 protocol down
ZXR10(config-event)#exit
ZXR10(config)#acl standard number 1
ZXR10(config-std-acl)#rule 1 permit any event zte
ZXR10(config-std-acl)#rule 2 deny any
ZXR10(config-std-acl)#exit
ZXR10(config)#interface gei_1/2
ZXR10(config-if)#ip access-group 1 in
```

When protocol on gei_1/1 is down, rule 1 becomes effective. Traffic can access gei_1/2. When protocol on gei_1/1 is up, rule 1 is not effective. Traffic fails to access gei_1/2 and can only access interface gei_1/1. In above cases, there is only one data flow can be received on SwitchC.

# Applying NP-Based ACL

ACLs that can be applied in NP mode include standard ACL, extended ACL, Layer 2 ACL, hybrid ACL, user-defined ACL, standard IPv6 ACL, extended IPv6 ACL and user-defined IPv6 ACL.

**Applying NP-Based ACL to Physical Port**

To apply NP-based ACL to physical port, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** <*interface-name*> | This enters interface configuration mode |
| 2 | ZXR10(config-if)#**ip access-group senior** <*acl-numbe* \| **acl name r**>{**in** \| **out**} | This applies NP-based ACL to physical port |

To cancel application of NP-based ACL to physical port, use **no ip access-group senior** <*acl-numbe* | **acl name r**>{**in** | **out**} command.

**Applying NP-Based ACL to VLAN**

To apply NP-based ACL to VLAN, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**vlan** <*vlan-number*> | This enters VLAN configuration mode |
| 2 | ZXR10(config-vlan)#**ip access-group senior** <*acl-numbe* \| **acl name r**>{**in** \| **out**} | This applies NP-based ACL to VLAN |

To cancel application of NP-based ACL to VLAN, use **no ip access -group senior** <*acl-numbe* | **acl name r**>{**in** | **out**} command.

**Applying NP-Based ACL to Smartgroup Interface**

To apply NP-based ACL to Smartgroup interface, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface smartgroup**<*number*> | This enters Smartgroup interface configuration mode |
| 2 | ZXR10(config-if)#**ip access-group senior** <*acl-numbe* \| **acl name r**>{**in** \| **out**} | This applies NP-based ACL to Smartgroup interface |

To cancel application of NP-based ACL to Smartgroup interface, use **no ip access-group senior** <*acl-numbe* | **acl name r**>{**in** | **out**} command.

# ACL Configuration Example

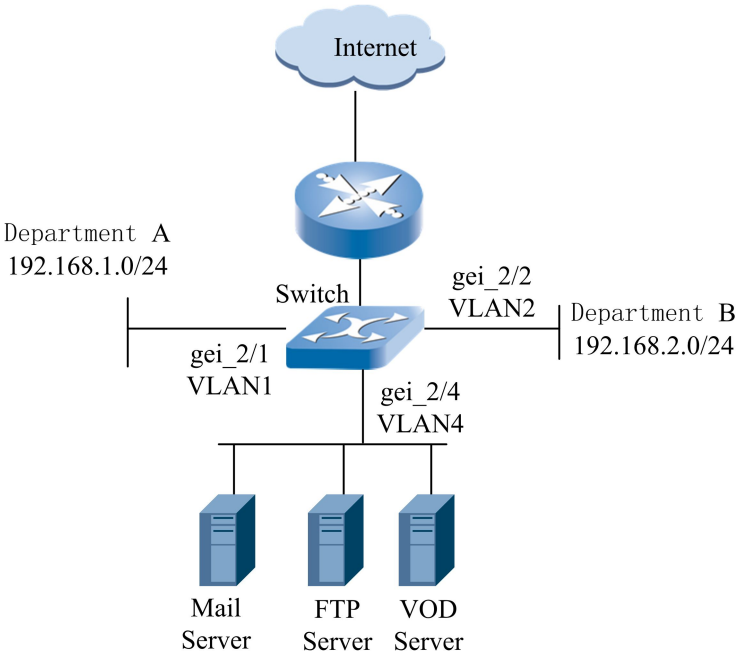
A company has an Ethernet switch, to which users of both A and B department and servers are connected. This is shown in Figure 26. The relevant provisions are as follows:

- Users of both A and B department are forbidden to access the FTP server and the VOD server in work time (9:00–17:00), but can access the Mail server at any time.
- Internal users can access the Internet through proxy 192.168.3.100, but users of department A are forbidden to access the Internet in work time.
- General Managers of both A and B department (with their IP addresses as 192.168.1.100 and 192.168.2.100 respectively) may access the Internet and all servers at any time.

The IP addresses of the servers are as follows:

- Mail server: 192.168.4.50
- FTP server: 192.168.4.60
- VOD server: 192.168.4.70

**FIGURE 26 ACL CONFIGURATION EXAMPLE**




Switch configuration:

```
/*Configure a time range*/
ZXR10(config)#time-range enable
ZXR10(config)#time-range working-time
ZXR10(config-tr)#periodic daily 09:00:00 to 17:00:00

/*Define an extended ACL to limit the users of Department A*/
ZXR10(config)#acl extend number 100
ZXR10(config-ext-acl)#rule 1 permit ip 192.168.1.100 0.0.0.0 any
ZXR10(config-ext-acl)#rule 2 deny ip 192.168.1.0 0.0.0.255 192
168.4.60 0.0.0.0 time-range working-time
ZXR10(config-ext-acl)#rule 3 deny tcp any eq 8888
```

```
192.168.4.70 0.0.0.0 time-range working-time
ZXR10(config-ext-acl)#rule 4 deny ip any 192.168.3.100 0.0.0.0
time-range working-time
ZXR10(config-ext-acl)#rule 5 permit ip any any

/*Define an extended ACL to limit the users of Department B */
ZXR10(config)#acl extend number 101
ZXR10(config-ext-acl)#rule 1 permit ip 192.168.2.100 0.0.0.0 any
ZXR10(config-ext-acl)#rule 2 deny ip 192.168.2.0 0.0.0.255
192.168.4.60 0.0.0.0 time-range working-time
ZXR10(config-ext-acl)#rule 3 deny tcp any eq 8888
192.168.4.70 0.0.0.0 time-range working-time
ZXR10(config-ext-acl)#rule 4 permit ip any any

/*Apply ACLs to the corresponding physical ports */
ZXR10(config)#interface fei_2/1
ZXR10(config-if)#ip access-group 100 in
ZXR10(config-if)#exit
ZXR10(config)#interface fei_2/2
ZXR10(config-if)#ip access-group 101 in
ZXR10(config-if)#exit
```

# ACL Maintenance and Diagnosis

To configure ACL maintenance and diagnosis, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show acl** [<*acl-number*>\|**name** <*acl-name*>] | This displays the contents of all ACLs or of the ACL with specified list number |
| 2 | ZXR10#**show running-config interface** <*port-name*> | This displays the configuration information of an Ethernet port |

This page is intentionally blank.

Chapter 10

# QoS Configuration

**Table of Contents**

QoS Overview ..... 91
Configuring QoS ..... 96
Configuring HQoS ..... 103
QoS Configuration Examples ..... 109
QoS Maintenance and Diagnosis ..... 111

## QoS Overview

Traditional network provides services at its best effort and all packets are treated in the same way. Network equipment sends messages to the destination in the principle of "first in first service" but does not guarantee transfer reliability and transfer delay of messages.

With the continuous emergence of new applications a new requirement for network service quality is raised because traditional network at the best effort cannot satisfy the requirement for applications. For example, user cannot use VoIP service and real-time image transmission normally if packet transfer delay is too long. To solve this problem, provide system with capability of supporting QoS.

**Functions**

When QoS is configured, it selects specific network traffic prioritizing it according to its relative importance and use. Implementing QoS in the network makes network performance more predictable and bandwidth utilization more effective. QoS provides the following functions:

- Traffic classification
- Traffic policing
- Traffic shaping
- Queue scheduling and default 802.1p
- Redirection and policy routing
- Priority marking
- Traffic mirroring
- Traffic statistics

# Traffic Classification

Traffic refers to packets passing through switch. Traffic classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet.

Traffic classification of QoS is based on ACL and the ACL rule must be permitted. The user can classify packets according to some filter options of the ACL which are as follows:

- Source IP address, destination IP address, source MAC address, destination MAC address, IP protocol type and TCP source port number
- TCP destination port number, UDP source port number, UDP destination port number, ICMP type, ICMP code, DSCP, ToS, precedence, source VLAN ID, Layer 2 Ethernet protocol type and 802.1p priority value

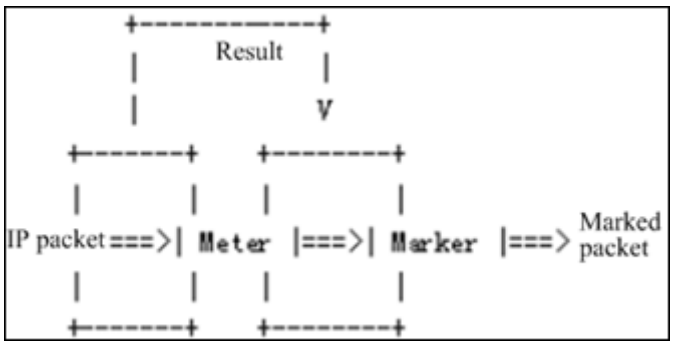
# Traffic Monitoring

Traffic monitoring involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are out of profile or nonconforming. Each policer specifies the action to take for packets that are in or out of profile. The following operations are specified by the policer:

- Discard or forward
- Change its DSCP value
- Change its discard priority (packets with the higher discard priority are discarded preferentially in case of queue congestion).

Traffic monitoring will not introduce extra delay and its working flow is shown in Figure 27.

**FIGURE 27 TRAFFIC MONITORING WORKING FLOW**




ZXR10 8900 series switch implements Single Rate Three Color Marker (SrTCM) (RFC2697) and Two Rate Three Color Marker (TrTCM) (RFC2698) functions, which both support color-blind and color-aware modes.

Meter works in two modes: color-blind mode and color-aware mode.

It assumes that packets are colorless in color-blind mode but assumes that packets are marked in a color in color-aware mode. A color is assigned to each packet passing through the switch according to a certain principle (packet information) on the switch. The Maker renders IP packets in the DS domain according to results given by the Meter.

Algorithm of the above two markers are described in details below.

**SrTCM** This algorithm is used in the Diffserv traffic conditioner to measure information flow and mark packets according to three traffic parameters (Committed Information Rate (CIR), Committed Burst Size (CBS) and Excess Burst Size EBS)). These parameters are called green, yellow and red markers. A packet is green if its size is less than CBS. A packet is yellow if its size is between CBS and EBS and is red if its size exceeds EBS.

**TrTCM** This algorithm is used in the Diffserv traffic conditioner to measure IP information flow and mark a packet in green, yellow or red according to the Peak Information Rate (PIR) and Committed Information Rate (CIR) and their relevant burst sizes (CBS and PBS). A packet is marked in red if its size exceeds PIR. A packet is marked in yellow if its size is between PIR and CIR and is marked in green if its size is less than CIR.

# Traffic Shaping

Traffic shaping is used to control the rate of output packets thus sending packets at even speed. Traffic shaping is used to match packet rate with downlink equipment to avoid congestion and packet discarding.

Traffic shaping is to cache packets whose rate exceeds the limited value and send packets at even rate; while traffic monitoring is to discard packets whose rate exceeds the limited value. Moreover, traffic shaping makes delay longer but traffic monitoring does not introduce any extra delay.

Traffic shaping is classified into the following two kinds:

- Incoming port bandwidth traffic shaping
- Outgoing port bandwidth traffic shaping

# Queue Scheduling and Default 802.1p

Each physical port of the ZXR10 8900 series switch supports eight output queues (queue 0 to queue 7) called CoS queues. Switch performs incoming port output queue operation according to the CoS queue corresponding to 802.1p of packets. In network congestion, the queue scheduling is generally used to solve the problem that multiple packets compete with each other for resources at the same time.

ZXR10 8900 series switch supports Strict Priority (SP), Weighted Round Robin (WRR) and Dynamic Weighted Round Robin (DWRR) queue scheduling modes. Eight output queues of a port can adopt different modes respectively.

**SP** SP is to strictly schedule data of each queue according to queue priority. First send packets in the highest priority queue and after that, send packets in the higher priority queue. Similarly, after that, send packets in the lower priority queue, and so on.

SP scheduling makes packets of key services processed preferentially, thus guaranteeing service quality of key services. But the low priority queue may never be processed and "starved".

**WRR** WRR makes each queue investigated possibly and not "starved". Each queue is investigated at different time, that is, has different weight indicating the ratio of resources obtained by each queue. Packets in the high priority queue have more opportunities to be scheduled than the low priority queue.

**DWRR** DWRR makes each queue investigated possibly. The weight of each queue is different. The difference between DWRR and WRR is that, the weight value of DWRR means the round scheduled bytes of eight queues on a port each time, in its unit of kbyte; while the weight value of WRR means the scheduled packet number of each queue. Therefore, DWRR does not effect much on bandwidth.

Data priority is contained in the 802.1P label. If data entering the port is not marked with an 802.1P label, a default 802.1p value will be assigned by the switch.

# Policy Routing

Redirecting is used to make the decision again about the forwarding of packets with certain features according to traffic classification. Redirection changes transmission direction of packets and export messages to the specific port, CPU or next-hop IP address.

Redirect packets to the next-hop IP address to implement policy routing.

On the aspect of packet forwarding control, policy-based routing has more powerful control capacity than traditional routing because it can select a forwarding path according to the matched field in the ACL. Policy routing can implement traffic engineering to a certain extent, thus making traffic of different service quality or different service data (such as voice and FTP) to go to different paths. The user has higher and higher requirements for network performance, therefore it is necessary to select different packet forwarding paths based on the differences of services or user categories.

# Priority Mark

Priority marking is used to reassign a set of service parameters to specific traffic described in the ACL to perform the following operations:

- Change the CoS queue of the packet and change the 802.1p value.
- Change the CoS queue of the packet and do not change the 802.1p value.
- Change the DSCP value of the packet.
- Change the discard priority of the packet.

# Traffic Mirroring

Traffic mirroring is used to copy a service flow matching the ACL rule to the CPU or specific port to analyze and monitor packets during network fault diagnosis.

# Traffic Statistics

Traffic statistics is used to sum up packets of the specific service flow. This is to understand the actual condition of the network and reasonably allocate network resources. The main content of traffic statistics contains the number of packets received from the incoming direction of the port.

# Queue-Based Bandwidth Upper and Lower Threshold

Due to limited queue buffer resources, when network congestion occurs, multiple packets will compete to use limited resources.

After configuring upper and lower threshold on outgoing interface and when multiple flows compete for limited resources, a cos queue flow can obtain a bandwidth which will not be less than bandwidth lower threshold or more than bandwidth upper threshold. In this way, no flow can occupy the entire bandwidth which makes the other flows fail to obtain any bandwidth.

# HQoS

Hierarchical QoS (HQoS) is to schedule and control traffic by configuring network topology extracted from actual network, which ensures quality of network.

**HQoS Functions**

HQoS has the following functions.

- Supporting hierarchical scheduling

  The most obvious characteristic of HQoS is hierarchical scheduling. It is used to simulate complex networks.

- Supporting mass of queues

  Different queues mean users of different services. HQoS can store packets received within 200ms at lone speed on a port. This can avoid congestion.

- Supporting mass of scheduling nodes

  Scheduling node is the main member to create topology model. It can express network topology factually. With the addition of scheduling hierarchy, the number of needed scheduling nodes will increase dramatically.

- Supporting good traffic monitoring and traffic control

  HQoS supports multiple traffic monitoring algorithms. It also supports configuration of CIR and PIR. Traffic less than CIR is guaranteed well. Traffic more than CIR and less than PIR is guaranteed when there is spare network bandwidth. CIR traffic and PIR traffic have different schedules.

# Configuring QoS

## Configuring Traffic Monitoring

To configure traffic monitoring, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**traffic-limit** <*acl-number*> **rule-id** <*rule-no*> **cir** <*cir-value*> **cbs** <*cbs-value*>{**ebs** <*ebs-value*>\|{**pir** <*pir-value*> **pbs** <*pbs-value*>}}{**mode** <*mode*>}[**drop-yellow**][**forward-red**][**remark-red-dp** {**high**\|**low**\|**medium**}][**remark-red-dscp**<*value*>][**remark-yellow-dp** {**high**\|**low**\|**medium**}][**remark-yellow-dscp** <*value*>] | This configures traffic monitoring |

**Note:**

Coloring algorithm is applied to traffic monitoring configuration. Parameters are described below.

| Parameter | Description |
|---|---|
| **ebs** | It means pbs parameter defined in protocol. |
| **pir** | It means using double rate marking algorithm. |
| **mode** | The value blind means switch works in color blindness mode. The value aware means switch works in color sensitivity mode. |
| **drop-yellow** | It means switch discards packets marked yellow. By default, switch transmits packets. |

| Parameter | Description |
|---|---|
| **forward-red** | It means switch transmits packets marked red. By default, switch discards packets. |
| **remark-red -dp** | It means remarking discarding priority of red packet. Priority parameters are high, medium and low. |
| **remark-red-d scp** | It means remarking DSCP priority of red packet. Priority parameters are 0 to 63. |
| **remark-yello w-dp** | It means remarking discarding priority of yellow packet. Priority parameters are high, medium and low. |
| **remark-yello w-dscp** | It means remarking DSCP priority of yellow packet. Priority parameters are 0 to 63. |

**Example** This example describes how to monitor and control traffic of packets with destination IP address 168.2.5.5 on port gei_5/1. Set the bandwidth to 10 M, burst transmission rate to no greater than 1M and change the DSCP value to 23 for the part that exceeds the limit and set the discard priority to high (this part of packets will be discarded at a higher priority in queue congestion).

```
ZXR10(config)#acl extend number 100
ZXR10(config-ext-acl)#rule 1 permit any 168.2.5.5
ZXR10(config-ext-acl)#exit
ZXR10(config)# traffic-limit 100 rule-id 1 cir 10000
cbs 2000 pir 10000 pbs 2000 mode blind
ZXR10(config)#interface gei_5/1
ZXR10(config-if)#ip access-group 100 in
```

# Configuring Traffic Rate Limit

To configure traffic rate limit, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-if)#**traffic-limit rate-limit** <*rate-value*> **bucket-size** <*value*>{**in**\|**out**} | This configures traffic rate limit |

**Example** This example describes how to enable traffic limit on gei_1/1. Configure egress rate to be 20M, and ingress rate to be 10M.

```
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#traffic-limit rate-limit 20000 bucket-size 4 out
ZXR10(config-if)#traffic-limit rate-limit 10000 bucket-size 4 in
```

# Configuring Layer 3 Rate Limit

To configure Layer 3 rate limit, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**nas** | This enters nas configuration mode |
| 2 | ZXR10(config-nas)#**ratelimit** | This enters ratelimit configuration mode |
| 3 | ZXR10(config-nas-ratelimit)#**ip host** *<ip-addr>* **vlan** *<vlan-id>*{**down-rate**\|**up-rate**}{**k***<64-1000>*\|**m***<10-1000>*} | This limits the rate of uplink or downlink users |
| 4 | ZXR10(config)#**show ratelimit** {**all**\|**host-ip** *<ip-addr>*} | This views configuration information of Layer 3 rate limit |

**Example** This example shows how to configure Layer 3 rate limit.

```
ZXR10(config)#nas
ZXR10(config-nas)#ratelimit
ZXR10(config-nas-ratelimit)#ip host 168.1.2.3 vlan 20 down-rate k 600
ZXR10(config-nas-ratelimit)#ip host 168.1.2.4 vlan 20 up-rate k 300
ZXR10(config-nas-ratelimit)#exit
ZXR10(config-nas)#exit
ZXR10(config)#show ratelimit all
Host-ip             Vlan     Up-rate   Down-rate
168.1.2.3           20          -       600K
168.1.2.4           20        300K        -
```

# Configuring Queue Scheduling

ZXR10 8900 series switch supports SP and WRR queue scheduling modes. When these two modes are mixed used, SP has a higher priority over WRR.

To configure queue scheduling, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-if)#**queue-mode** {**strict-priority**\|{**dwrr** *<queue-no><dwrr-weight>***&***<1-8>*}\|{**wrr** *<queue-no><wrr-weight>***&***<1-8>*}} | This configures queue scheduling and default 802.1p priority on port. |

**Note:**

Value range of *dwrr-weight* is 1~160000. Value range of *wrr-weight* is 1~15.

**Example** Configure strict scheduling based on priority on interface gei_1/1. Enable WRR scheduling on interface gei_1/2. Weights of Queues 0~7 are 10, 5, 8, 10, 5, 8, 9, 10 respectively. Set the default 802.1p of interface gei_1/2 to 5.

```
ZXR10(config)#interface gei_1/1
ZXR10(config-gei_1/1)#queue-mode strict-priority
ZXR10(config-gei_1/1)#exit
ZXR10(config)#interface gei_1/2
```

```
ZXR10(config-gei_1/2)#queue-mode wrr 0 10
ZXR10(config-gei_1/2)#queue-mode wrr 1 5
ZXR10(config-gei_1/2)#queue-mode wrr 2 8
ZXR10(config-gei_1/2)#queue-mode wrr 3 10
ZXR10(config-gei_1/2)#queue-mode wrr 4 5
ZXR10(config-gei_1/2)#queue-mode wrr 5 8
ZXR10(config-gei_1/2)#queue-mode wrr 6 9
ZXR10(config-gei_1/2)#queue-mode wrr 7 10
ZXR10(config-gei_1/2)#priority 5
```

# Configuring Policy Routing

To configure policy routing, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**redirect in** <*acl-number*> **rule-id** <*rule-no*>{**cpu** \|{**interface** <*port-name*>}\|{**next-hop1** <*ip-address*><*priority*>}} | This configures policy routing. |

**Example** This example shows how to redirect packet. Redirect packets with source IP address 168.2.5.5 on gei_1/4 to gei_1/3. Designate the next hop IP address 166.88.96.56 to packets with destination address 66.100.5.6.

```
ZXR10(config)#acl extended number 100
ZXR10(config-ext-acl)#rule 1 permit ip 168.2.5.5 0.0.0.0 any
ZXR10(config-ext-acl)#rule 2 permit ip any 66.100.5.6 0.0.0.0
ZXR10(config-ext-acl)#exit
ZXR10(config)#redirect in 100 rule-id 1 interface gei_1/3
ZXR10(config)#redirect in 100 rule-id 2 next-hop1 166.88.96.56 1
ZXR10(config)#interface gei_1/4
ZXR10(config-if)#ip access-group 100 in
```

# Configuring Priority Mark

To configure priority marking, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**priority-mark** <*acl-number*> **rule-id** <*rule-no*>{[**dscp** <*dscp-value*>][**drop-precedence** <*drop-value*>][**cos** <*cos-value*>\|**local-precedence** <*local-value*>][**out-vlanID** <*vlan-id*>][**precedence** <*precedence-value*>] | This configures priority marking |

**Example** This example describes how to change DSCP value of packets with source IP address 168.2.5.5 on port gei_5/1 to 34, and select 4 for output queues.

```
ZXR10(config)#acl basic number 10
ZXR10(config-basic-acl)#rule 1 permit 168.2.5.5
ZXR10(config-basic-acl)#exit
ZXR10(config)#priority-mark 10 rule-id 1 dscp 34 cos 4
ZXR10(config)#interface gei_5/1
ZXR10(config-if)#ip access-group 10 in
```

# Configuring Tail Discarding

To configure tail discarding, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**qos tail-drop** *<session-index>* **queue-id** *<queue-id><green-threshold><yellow-threshold><red-threshold>* | This configures parameters of packets to be discarded |
| 2 | ZXR10(config)#**interface** *<interface-name>* | This enters interface configuration mode |
| 3 | ZXR10(config-if)#**drop-mode tail-drop** *<session-index>* | This discards packets |

**Example** This example shows how to configure tail discarding. Configure tail discarding function on gei_1/1. Yellow packets with waterline 100, red packets with waterline 120 and green packets with waterline 120 are discarded.

```
ZXR10(config)#qos tail-drop 1 queue-id 1 120 100 120
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#drop-mode tail-drop 1
```

# Configuring COS Discarding Priority Mapping

To configure COS discarding priority mapping, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**qos cos-drop-map** *<cos-0-drop-priority><cos-1-drop-priority><cos-2-drop-priority><cos-3-drop-priority><cos-4-drop-priority><cos-5-drop-priority><cos-6-drop-priority><cos-7-drop-priority>* | This configures parameters of COS discarding priority |
| 2 | ZXR10(config)#**interface** *<interface-name>* | This enters interface configuration mode |
| 3 | ZXR10(config-if)#**trust-cos-drop enable** | This applies COS discarding priority mapping function |

**Note:**

To disable COS discarding priority mapping function, use **trust-cos-drop disable** command.

**Example** This example shows how to configure COS discarding priority mapping. Configure COS discarding priority mapping on gei_1/1. Priority of queue 7 is high, other priorities are low.

```
ZXR10(config)#qos cos-drop-map 1 1 1 1 1 1 1 2
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#trust-cos-drop enable
```

# Configuring COS Local Priority Mapping

To configure COS local priority mapping function, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**qos cos-local-map** *<cos-0-local-priority><cos-1-local-priority><cos-2-local-priority><cos-3-local-priority><cos-4-local-priority><cos-5-local-priority><cos-6-local-priority><cos-7-local-priority>* | This configures parameters of COS local priority |
| 2 | ZXR10(config)#**interface** *<interface-name>* | This enters interface configuration mode |
| 3 | ZXR10(config-if)#**trust-cos-local enable** | This applies COS local priority mapping function |

**Note:**

To disable COS local priority mapping function, use **trust-cos-local disable** command.

**Example** This example shows how to configure COS local priority mapping. Configure COS local priority mapping on gei_1/1. Priority of queue 1 is 1, priority of queue 2 is 2, and the rest are deduced by analogy.

```
ZXR10(config)#qos cos-local-map 1 2 3 4 5 6 7
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#trust-cos-local enable
```

# Configuring DSCP Priority Mapping

To configure DSCP priority mapping, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**qos conform-dscp** <*dscp-list*><*dscp-value*><*cos-value*><*drop-priority*> | This configures DSCP priority mapping. |
| 2 | ZXR10(config)#**interface** <*interface-name*> | This accesses L2 configuration interface. |
| 3 | ZXR10(config-if)#**trust-dscp enable** | This applies DSCP priority mapping. |

By executing command **trust-dscp disable**, DSCP priority mapping can be cancelled.

**Example** This example shows how to configure DSCP priority mapping on interface gei_1/1. Map DSCP value 30 to 20 and set COS value to 0 and drop priority to high.

```
ZXR10(config)#qos conform-dscp 30 20 0 2
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#trust-dscp enable
```

# Configuring Traffic Mirroring

To configure traffic mirroring, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**traffic-mirror in** <*acl-number*> **rule-id** <*rule-no*>{**cpu**\|**interface** <*port-name*>} | This configures traffic mirroring |

**Example** This example describes how to map data traffic with source IP address 168.2.5.6 on port gei_1/8 to port gei_1/4.

```
ZXR10(config)#acl basic number 10
ZXR10(config-basic-acl)#rule 1 permit 168.2.5.5
ZXR10(config-basic-acl)#rule 2 permit 168.2.5.6
ZXR10(config-basic-acl)#exit
ZXR10(config)#traffic-mirror in 10 rule-id 2 interface
ZXR10(config)#interface gei_1/8
ZXR10(config-if)#ip access-group 10 in
ZXR10(config-if)#exit
ZXR10(config)#interface gei_1/4
ZXR10(config-if)#monitor session 1 destination
```

# Configuring Traffic Statistics

To configure traffic statistics, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**traffic-statistics** <*acl-number*> **rule-id** <*rule-no*> **pkt-type** {**all**\|**green**\|**red**\|**yellow**} **statistics-type** {**byte**\|**packet**} | This configures traffic statistics |

**Example** This example describes how to collect traffic statistics on data in the network with destination IP address 67.100.88.0/24 on port gei_4/8.

```
ZXR10(config)#acl extend number 100
ZXR10(config-ext-acl)#rule 1 permit ip 168.2.5.5 0.0.0.0 any
ZXR10(config-ext-acl)#rule 2 permit ip any 67.100.88.0 0.0.0.255
ZXR10(config-ext-acl)#exit
ZXR10(config)#traffic-statistics in 100 rule-id 2
ZXR10(config)#interface gei_4/8
ZXR10(config-if)#ip access-group 100 in
```

## Configuring Queue-Based Bandwidth Upper and Lower Threshold

| Step | Command | Functions |
|---|---|---|
| 1 | ZXR10(config)#**interface** < *interface-name*> | This accesses L2 configuration interface. |
| 2 | ZXR10(config-if)#**traffic-shape** { **queue** <*queue-number*>{[**max-datarate-limit** <*rate*>]|[**min-gua-datarate** <*rate*>]}} | This configures queue-based bandwidth upper and lower threshold. |

# Configuring HQoS

## Configuring Traffic Class

To configure traffic class, perform the following steps.

1. To create a traffic class or enter a traffic class, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**flow-class** <*class-name*> | This creates a traffic class or enters a traffic class |

To delete a traffic class, use **no flow-class** <*class-name*> command. If the traffic class is used, the class can not be deleted.

2. To configure a matching rule, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-fclass)#**match** {**(acl** <*acl-no*> **rule** <*rule-no*>**)** \| **tunnel** <*1-4096*>\| **vlan** <*1-4094*>\| **vip** <*1-16384*>}\| **phb** {**be** \| **af1** \| **af2** \| **af3** \| **af4** \| **ef** \| **cs6** \| **cs7**}} | This configures a matching rule in traffic class configuration mode |

One traffic class can only match one ACL rule. If an ACL rule matches flow-class, the class must exist and the class can not be deleted. Corresponding ACL and rule number must exist.

To delete a ACL rule, use **no match** {**acl** <*acl-no*> **rule** <*rule -no* | **tunnel** <*tunnel-no*>| **flow-class** <*class-name*>} command.

3. To display traffic class information, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**show flow-class** [<*class-name*>] | This displays traffic class information |

If class name is not configured, information of all traffic classes is displayed.

**Example** This example shows view traffic class information.

```
ZXR10(config)#show flow-class voice
Flow-class void
Match acl 1 rule 1
Match acl 1 rule 3
```

# Configuring WRED Policy

To configure WRED policy, perform the following steps.

1. To create or enter a WRED policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**wred-profile** <*profile-name*>[**level** <*1-3*>] | This creates or enters a WRED policy |

Instructions:

- Users enter WRED policy view after inputting this command. If the policy does not exist, users should input **level** to create a policy.
- Each level has a default WRED. They are default1, default2 and default3.
- By default, level 1 can be configured up to 32 policies, level 2 can be configured up to 32 policies, and level 3 can be configured up to 8 policies.

To delete a WRED policy, use **no wred-profile** <*profile-name*> command.

In global configuration mode, if a view is used, this view can not be deleted. Default1, default2 and default3 can not be deleted.

2. To configure discarding parameters of WRED policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-wred)#**color** {**red** \| **yellow** \| **green**} **min** <*0-256000*> **max** <*20-256000*> **percent** <*0-100*> | This configures discarding parameters of WRED policy. |

By default, the minimum and maximum values of red, yellow and green are 100, and the value of percent is 0.

# Configuring WFQ Policy

To configure WFQ policy, perform the following steps.

1. To create or enter a WFQ policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**wfq-profile** <*profile-name*>[**level** <*1-3*>] | This creates or enters a WFQ policy |

Instructions:

- Users enter WFQ policy view after inputting this command. If the policy does not exist, users should input **level** to create a policy.
- Each level has a default WFQ. They are default1, default2 and default3.
- By default, level 1 can be configured up to 64 policies, level 2 can be configured up to 64 policies, and level 3 can be configured up to 16 policies.

To delete a WFQ policy, use **no wfq-profile** <*profile-name*> command.

In global configuration mode, if a view is used, this view can not be deleted. Default1, default2 and default3 can not be deleted.

2. To configure discarding parameters of WFQ policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-wfq)#**weight** <*1-256*> | This configures discarding parameters of WFQ policy. |

By default, the weight is 1.

# Configuring Traffic Shaping

To configure traffic shaping policy, perform the following steps.

1. To create or enter a traffic shaping policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**shaping-profile** <*profile-name*>[**level** <*2-4*>] | This creates or enters a traffic shaping policy |

Instructions:

- Users enter traffic shaping policy view after inputting this command. If the policy does not exist, users should input **level** to create a policy.
- Each level has a default shaping. They are default2 , default3 and default 4..
- By default, level 2 can be configured up to 254 policies, level 3 can be configured up to 15 policies and level 4 can be configured up to 31 policies.

To delete a WRED policy, use **no shaping-profile** <*profile-name*> command.

In global configuration mode, if a view is used, this view can not be deleted. Default1, default2 and default3 can not be deleted.

2. To configure discarding parameters of traffic shaping policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-shaping)#**cir** <*1-10000000*> **cbs** <*1024-16711680*> **pir** <*1-10000000*> **pbs** <*1024-16711680*> | This configures discarding parameters of traffic shaping policy. |

By default, the value of CIR and PIR is 1.

# Configuring HQoS Policy

To configure HQoS policy, perform the following steps.

1. To enter policy view, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**qos-policy** <*policy-name*>[**level** <*1-3*> **mode** {**TUNNEL** | **VLAN**}] | This enters policy view |

If the policy does not exist, users should input **level** to create a policy. The policy name is within 32 characters.

To delete a policy, use **no qos-policy** <*policy-name*> command.

2. To configure policy description, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy)#**description** <*string*> | This configures policy description. The description is within 200 characters |

To delete policy description, use **no description** command.

3. To enter traffic class, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy)#**flow-class** <*class-name*> | This enters traffic class |

Each policy has a default traffic class named class default. WRED, WFQ and shaping of the default traffic class can be configured.

4. To configure queue priority, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy-class)#**priority** {**high** \| **low**} | This configures queue priority |

5. To apply WFQ policy to a traffic class, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy-class)#**wfq-profile** <*profile-name*> | This applies WFQ policy to a traffic class |

By default, a traffic class is associated with a default WFQ policy of corresponding level. If the WFQ policy does not exist, system prompts error.

To cancel WFQ policy of a traffic class, use **no wfq-profile** command.

6. To apply WRED policy to a traffic class, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy-class)#**wred-profile** <*profile-name*> | This applies WRED policy to a traffic class |

By default, a traffic class is associated with a default WRED policy of corresponding level.

To cancel WRED policy of a traffic class, use **no wred-profile** command.

7. To apply shaping policy to a traffic class, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy-class)#**shaping-profile** <*profile-name*> | This applies shaping policy to a traffic class |

By default, a traffic class is associated with a default shaping policy of corresponding level. Traffic class of level 1 can not be associated with a shaping policy.

To cancel shaping policy of a traffic class, use **no shaping-pr ofile** command.

8. To apply sub-policy to a traffic class, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-qpolicy-class)#**policy** <*policy-name*> | This applies sub-policy to a traffic class. The level of sub-policy should be lower |

9. To apply policy to an interface, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-if)#**qos-policy** <*policy-name*>{**in** \| **out**} **shaping** <*shaping-name*> | This applies policy to an interface. The interface can be a physical port, a Layer 2 VLAN port or a Smartgroup interface. |

10. To copy QoS policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**copy qos-profile source** <*profile-name*> **destination** <*profile-name*>[**overwrite**] | This copies QoS policy |

If the source policy does not exist, system prompts error. If policy name in destination has existed, and users do not set the covering mode, system prompts error.

11. To display policy, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**show qos-policy** [<*policy-name*>[**detail**]] | This displays policy |

When the policy name is not configured, information of all policies is displayed. If a policy name is configured, information of its sub-policy is also displayed.

12. To display policy statistic information on an interface, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**show qos-policy statistics** {**interface** <*name*>\| **vlan** <*vlan-id*>}{**in** \| **out**} | This displays policy statistic information on an interface |

13. To clear policy statistic information on an interface, use the following command.

| Command | Function |
|---|---|
| ZXR10(config-if)#**clear qos-policy statistics** {**in** \| **out**} | This clears policy statistic information on an interface |

**Example** This example shows detailed statistic information of policy named telecom.

```
ZXR10 #show qos-policy telcom detail
Qos-policy telcom:
Class voice
Match acl 1 rule 1
Class video
Match acl 1 rule 3
Policy video
  Class CCTV1
   Match acl 1 rule 5
```

This example shows policy statistic information on gei_2/1.

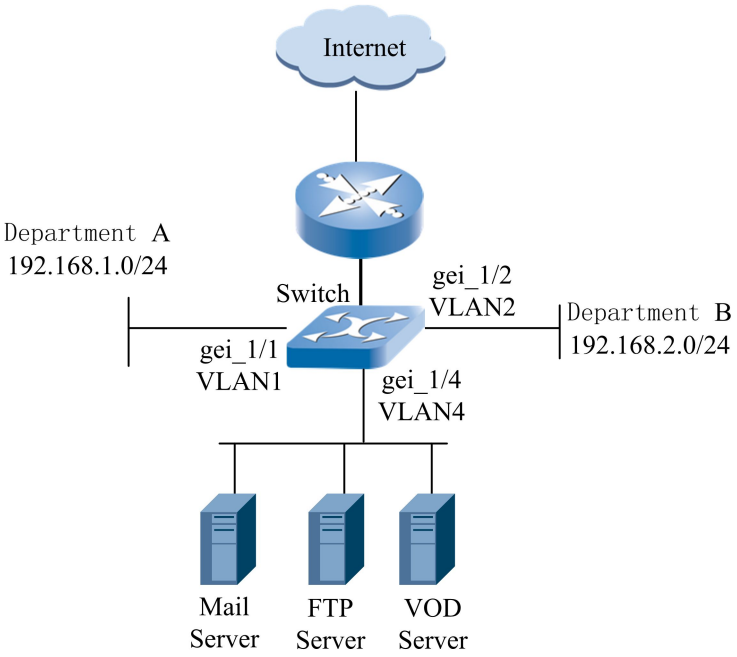
```
ZXR10 #show qos-policy statistics interface gei_2/1 in
Qos-policy telcom:
Class voice
Receive Packet:10000
Reveive byte: 1000000
Drop packet:100
Drop byte:10000
Class video
```

# QoS Configuration Examples

## Typical QoS Configuration Example

Network A, Network B and internal servers are connected to an Ethernet switch, as shown in Figure 28. Internal servers include a VOD server with IP address 192.168.4.70. To ensure QoS of VOD, it should be configured with a higher priority. Internal users can access Internet through proxy 192.168.3.100. However, bandwidth of Network A and B should be limited and traffic statistics is required.

**FIGURE 28 TYPICAL QOS CONFIGURATION EXAMPLE**




Configuration on the switch:

```
ZXR10(config)#acl extended number 100
ZXR10(config-ext-acl)#rule 1 permit tcp any 192.168.4.70 0.0.0.0
ZXR10(config-ext-acl)#rule 2 permit ip any 192.168.3.100 0.0.0.0
ZXR10(config-ext-acl)#rule 3 permit ip any any
ZXR10(config-ext-acl)#exit

ZXR10(config)#priority-mark 100 rule-id 1 dscp 62 cos 7
/*To ensure the QoS of VOD, change the 802.1p value to 7*/

ZXR10(config)#traffic-limit 100 rule-id  2 cir 5000 cbs 2000
ebs 3000 mode blind
/*Limit the bandwidth of the access from Network A to the Internet*/

ZXR10(config)#traffic-statistics 100 rule-id 2 pkt-type all
statistics-type byte
/*Collect the statistics on the traffic of Network A*/

ZXR10(config)#interface gei_1/1
ZXR10(config-if)#ip access-group 100 in
ZXR10(config-if)#exit
/*Apply ACL 100 to the interface connecting to Network A*/

ZXR10(config)#acl extended number 101
ZXR10(config-ext-acl)#rule 1 permit tcp 192.168.2.0 0.0.0.255
192.168.4.70 0.0.0.0
ZXR10(config-ext-acl)#rule 2 permit ip any 192.168.3.100 0.0.0.0
ZXR10(config-ext-acl)#rule 3 permit ip any any
ZXR10(config-ext-acl)#exit

ZXR10(config)#priority-mark 101 rule-id 1 dscp 62 cos 7
/*To ensure the QoS of VOD, change the 802.1p value to 7*/

ZXR10(config)#traffic-limit 101 rule-id 2 cir 10000 cbs 2000
ebs 3000 mode blind
/*Limit the bandwidth of the access from Network B to the Internet*/

ZXR10(config)#traffic-statistics 101 rule-id 2 pkt-type all
statistics-type byte
/*Collect the statistics on the traffic of Network B*/

ZXR10(config)#interface gei_1/2
```

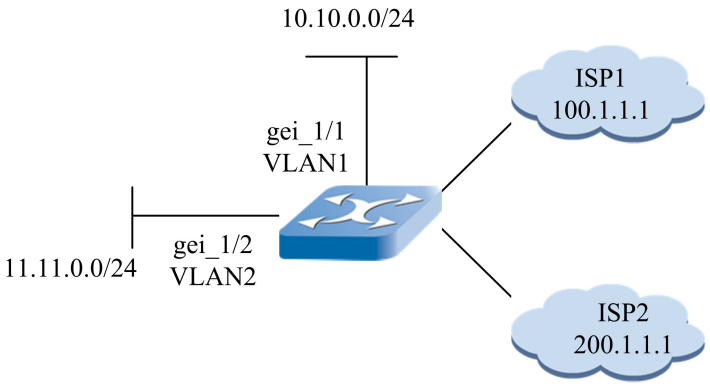
```
ZXR10(config-if)#ip access-group 101 in
/*Apply ACL 101 to the interface connecting to Network B*/
```

# Policy Routing Configuration Example

When multiple Internet service provider (ISP) egresses exist in a network, different ISP egresses can be selected for different groups of users by policy routing.

As shown in Figure 29, select different egresses according to the IP addresses of users. Users in sub-network 10.10.0.0/24 use the ISP1 egress. Users in sub-network 11.11.0.0/24 use the ISP2 egress.

**FIGURE 29 POLICY ROUTING CONFIGURATION EXAMPLE**




Configuration of switch:

```
ZXR10(config)#acl standard number 10
ZXR10(config-std-acl)#rule 1 permit 10.10.0.0 0.0.0.255
ZXR10(config-std-acl)#rule 2 permit 11.11.0.0 0.0.0.255
ZXR10(config-std-acl)#exit
ZXR10(config)#redirect in 10 rule-id 1 next-hop 100.1.1.1
ZXR10(config)#redirect in 10 rule-id 2 next-hop 200.1.1.1
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#ip access-group 10 in
ZXR10(config-if)#exit
ZXR10(config)#interface gei_1/2
ZXR10(config-if)#ip access-group 10 in
```

# QoS Maintenance and Diagnosis

To configure QoS maintenance and diagnosis, use the following command.

| Command | Function |
| --- | --- |
| ZXR10(config)#**show qos** [**name** <*acl-name*>| **number** <*acl-number*>] | This views QoS configuration information |

**Example** This example shows how to view QoS configuration information.

```
ZXR10(config)#acl standard number 1
ZXR10(config-std-acl)#rule 1 permit 100.1.1.1
ZXR10(config-std-acl)#exit
ZXR10(config)#traffic-limit 1 rule-id 1 cir 10000 cbs 2000
ebs 2000 mode blind
ZXR10(config)#show qos

traffic-limit 1 rule-id 1 cir 10000 cbs 2000 ebs 2000 mode blind
```

Chapter 11

# DOT1x Configuration

**Table of Contents**

DOT1x Overview ..... 113
Configuring DOT1x ..... 114
DOT1x Configuration Examples..... 117
DOT1x Maintenance and Diagnosis..... 120

## DOT1x Overview

DOT1X is IEEE 802.1x, is a port-based network access control protocol. It optimizes the authentication mode and authentication architecture and solves the problems caused by traditional PPPoE and Web/Portal authentication modes; therefore it is more suitable for the broadband Ethernet.

IEEE 802.1x protocol architecture contains three major parts: supplicant system, authenticator system and authentication server system.

**Supplicant System**

Client system is a user terminal system where client software is often installed. User originates IEEE802.1x protocol authentication by booting the client software. To support port-based access control, the client system needs to support the Extensible Authentication Protocol Over LAN (EAPOL).

**Authentication System**

Authentication system is network equipment supporting the IEEE802.1x protocol, such as the switch. Corresponding to every different user port (physical port or MAC address, VLAN and IP of the user equipment), the equipment has two logical ports composed of the controlled port and uncontrolled port.

Uncontrolled port is always in bidirectional connection state and delivers EAPOL protocol frames thus ensuring the client to always send or receive authentication.

Controlled port opens upon success of the authentication and delivers network resources and services. The controlled port modes can be configured as bidirectional control and only in direction control to adapt to different application environments. When the user fails to pass authentication, the controlled port is in unauthenticated state and the user cannot access services offered by the authentication system.

Controlled and uncontrolled ports in the IEEE 802.1x protocol are logical concepts and such physical switches are inexistent in the equipment. The IEEE 802.1x protocol establishes a logical au-

thentication channel for each user and other users cannot use the logical channel after the port is enabled.

**Authentication Server System**

Authentication server is usually a RADIUS server. In authentication server user-related information is stored such as the VLAN where the user locates, CAR parameter, priority and access control list of the user. Once the user passes authentication, the authentication server delivers user-related information to the authentication system which creates a dynamic access control list. The above parameters are used to measure subsequent traffic of the user. Authentication server and RADIUS server communicate with each other through the RADIUS protocol.

# Configuring DOT1x

## Configuring AAA

To configure AAA, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**nas** | This enters nas configuration mode |
| 2 | `ZXR10(config-nas)#`**create aaa** <*rule-id*>[**port** <*port-name*>][**vlan** <*vlan-id*>] | This creates AAA control entry |
| 3 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **control** {**dot1x**\|**dot1x-relay**}{**enable**\|**disable**} | This enables/disables dot1x authentication or relay |
| 4 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **authentication** {**auto**\|**locl**\|**radius**} | This selects an authentication mode |
| 5 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **protocol** {**pap**\|**chap**\|**eap**} | This selects an authentication protocol |
| 6 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **keepalive** {**enable** [**period** <*period-value*>]\|**disable**} | This configures keepalive interval |
| 7 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **accounting** {**enable**\|**disable**} | This configures to charge or not |
| 8 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **multiple-hosts** {**enable** [**max-hosts** <*host-number*>]\|**disable**} | This configures whether multiple users are allowed or not and configures user quota |
| 9 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **default-isp** <*isp-name*> | This configures the default ISP server name |
| 10 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **fullaccount** {**enable**\|**disable**} | This configures whether to contain ISP domain name in user name |
| 11 | `ZXR10(config-nas)#`**aaa** <*rule-id*> **groupname** <*group-name*> | This configures a group name |

| Step | Command | Function |
|---|---|---|
| 12 | ZXR10(config-nas)#**aaa** <*rule-id*> **radius-server** [**accounting** \| **authentication**]<*group-number*> | This binds an AAA control entry with the radius server group |
| 13 | ZXR10(config-nas)#**aaa** <*rule-id*> **authorization** {**auto**\|**unauthorized**\|**authorized**} | This configures the authorization mode |

**Note:**

To clear an AAA control entry, use **clear aaa** <*rule-id*> command.

# Configuring DOT1x Parameters

To configure DOT1x, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**nas** | This enters nas configuration mode |
| 2 | ZXR10(config-nas)#**dot1x re-authentication** {**enable** [**period** <*period*>]\|**disable**} | This configures dot1x re-authentication cycle |
| 3 | ZXR10(config-nas)#**dot1x quiet-period** <*period*> | This configures quiet period of dot1x authentication |
| 4 | ZXR10(config-nas)#**dot1x tx-period** <*period*> | This sets seconds for timeout and resending request for authentication |
| 5 | ZXR10(config-nas)#**dot1x supplicant-timeout** <*period*> | This configures online detection timeout time of the dot1x user |
| 6 | ZXR10(config-nas)#**dot1x server-timeout** <*period*> | This configures the timeout of the dot1x authentication |
| 7 | ZXR10(config-nas)#**dot1x max-requests** <*count*> | This configures maximum request times of dot1x authentication |

# Configuring Local Authentication User

To configure local authentication user, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**nas** | This enters nas configuration mode |
| 2 | ZXR10(config-nas)#**create localuser** <*user-id*>[**name** <*user-name*>][**password** <*user-password*>] | This creates a local user |
| 3 | ZXR10(config-nas)#**localuser** <*user-id*> **port** <*port-name*> | This binds the user with the port |
| 4 | ZXR10(config-nas)#**localuser** <*user-id*> **vlan** <*vlan-id*> | This binds the user with VLAN |
| 5 | ZXR10(config-nas)#**localuser** <*user-id*> **mac** <*mac-address*> | This binds the user with MAC address |
| 6 | ZXR10(config-nas)#**localuser** <*user-id*> **accounting** {**enable**\|**disable**} | This configures accounting attribute of users |

**Note:**

To delete a local user, use **clear localuser** <*user-id*> command.

# Managing DOT1x Authentication User

To manage access users of DOT1x authentication, perform the following steps.

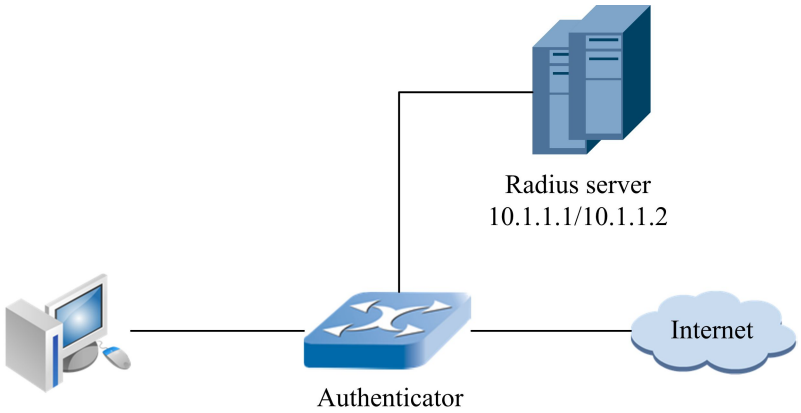
| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**show client** {{**port** <*port-number*>[**vlan** <*vlan-number*>]}\|{**slot** <*slot-number*> **index** <*index-number*>}\| **statistics**} | This displays all dot1x authenticated users |
| 2 | ZXR10(config-nas)#**clear client** [{**slot** <*slot-number*> **index** <*index-number*>}\|**port** <*port-name*>\| **vlan** <*vlan-id*>] | This deletes a specified user |

# DOT1x Configuration Examples

## Dot1x Radius Authentication Application

Workstation of a user is connected to Ethernet A of the Ethernet switch. This is shown in Figure 30.

**FIGURE 30 DOT1X RADIUS AUTHENTICATION APPLICATION**




The following procedures are required to be implemented on the switch:

- Conduct user access authentication on each port to control the user's access to the Internet.
- It is required that the access control mode is MAC address-based access control mode.
- All AAA access users belong to the default domain zte163.net.
- This authentication and RADIUS authentication are conducted at the same time.
- Disconnect the user and make it offline if RADIUS accounting fails.
- Do not add the domain name after the user name during access.
- Connect the server group composed of two RADIUS servers to the switch. IP addresses of these servers are 10.1.1.1 and 10.1.1.2 respectively. It is required that the former serves as the master authentication/slave accounting server and the latter serves as the slave authentication/master accounting server.
- Set the encryption key to be "aaazte" when the system exchanges packets with the authentication RADIUS server. Set the system to resend packets to the RADIUS server if no response comes from this server within five seconds after the

previous sending, and packets can be resent for five times at most. Direct the system to remove the user domain name from the user name and before sending it to the RADIUS server.

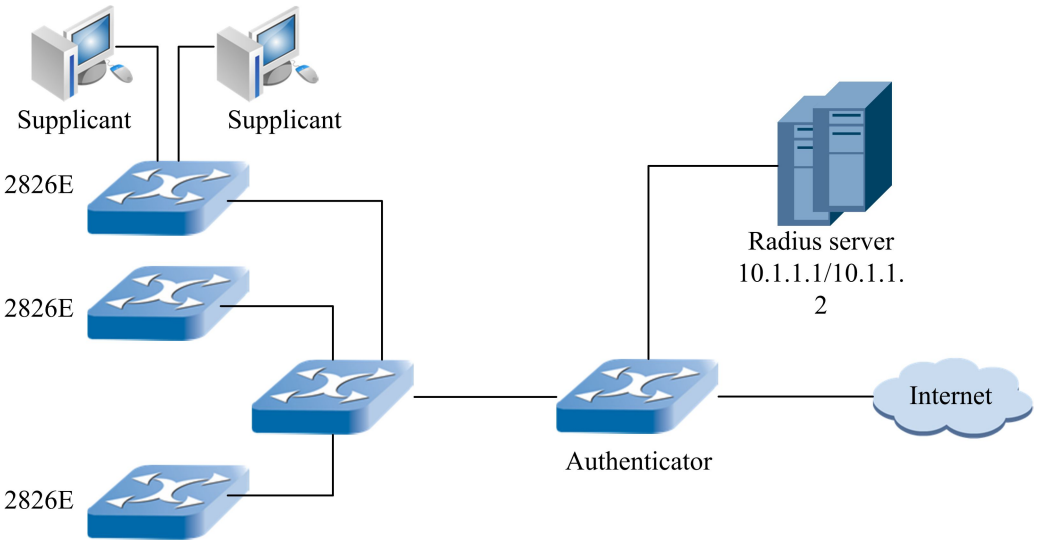
Configuration on the switch:

```
ZXR10(config)#radius authentication-group 1
ZXR10(config-authgrp-1)#server 1 10.1.1.1 master key aaazte
port 1812
ZXR10(config-authgrp-1)#server 2 10.1.1.2 key aaazte port 1812
ZXR10(config-authgrp-1)#max-retries 5
ZXR10(config-authgrp-1)#timeout 5
ZXR10(config-authgrp-1)#exit
ZXR10(config)#radius accounting-group 1
ZXR10(config-acctgrp-1)#server 1 10.1.1.2 master key aaazte
port 1813
ZXR10(config-acctgrp-1)#server 2 10.1.1.1  key aaazte port 1813
ZXR10(config)#nas
ZXR10(config-nas)#create aaa 1 port fei_1/1
ZXR10(config-nas)#aaa 1 control dot1x enable
ZXR10(config-nas)#aaa 1 authorization auto
ZXR10(config-nas)#aaa 1 accounting enable
ZXR10(config-nas)#aaa 1 multiple-hosts enable
ZXR10(config-nas)#aaa 1 default-isp zte163.net
ZXR10(config-nas)#aaa 1 fullaccount disable
ZXR10(config-nas)#aaa 1 radius-server authentication 1
ZXR10(config-nas)#aaa 1 radius-server accounting 1
```

# Dot1x Relay Authentication Application

Intranet topology of an enterprise is shown in Figure 31.

**FIGURE 31 DOT1X RELAY AUTHENTICATION APPLICATION**




The criterion is that only the authorized hosts are granted access to the Internet resources while the others can only get access to the Intranet resources.

- Divide hosts in the enterprise into a sub-network (or multiple sub-networks), where the hosts can access each other.

- Enable 802.1X relay function on Ethernet switch inside sub-network and enable 802.1X authentication on Ethernet port of the sub-network gateway.
- Do not charge users inside enterprise, and only authenticate them on the Radius server. Master/slave authentication servers are 10.1.1.1/10.1.1.2 respectively. It is assumed that enterprise uses 2826E Ethernet switch inside it and uses ZXR10 8905 Ethernet switch as the gateway.

Configuration on 2826E:

```
Set dot1xreley enable
```

Configuration on ZXR10 8905:

```
ZXR10(config)#radius authentication-group 1
ZXR10(config-authgrp-1)#server 1 10.1.1.1 master key aaazte
port 1812
ZXR10(config-authgrp-1)#server 2 10.1.1.2 key aaazte port 1812
ZXR10(config-authgrp-1)#exit
ZXR10(config)#nas
ZXR10(config-nas)#create aaa 1 port fei_1/1
ZXR10(config-nas)#aaa 1 control dot1x enable
ZXR10(config-nas)#aaa 1 authorization auto
ZXR10(config-nas)#aaa 1 accounting disable
ZXR10(config-nas)#aaa 1 multiple-hosts enable
ZXR10(config-nas)#aaa 1 default-isp zte163.net
ZXR10(config-nas)#aaa 1 fullaccount disable
ZXR10(config-nas)#aaa 1 radius-server authentication 1
```

# Dot1x Local Authentication Application

In the applications of Dot1x radius authentication and Dot1x relay authentication, enterprise wants to register network card address of each host. When user logs in from the dot1x client, only MAC address of the network card is checked. User can log in only when address is legal.

Enterprise numbers for each MAC address and Internet access duration of the user is based on the number. A ZXR10 8908 switch works as the authenticator and it can implement the application requirement. The application configuration is shown below.

```
ZXR10(config)#nas
ZXR10(config-nas)#create aaa 1 port fei_1/1
ZXR10(config-nas)#aaa 1 control dot1x enable
ZXR10(config-nas)#aaa 1 authorization auto
ZXR10(config-nas)#aaa 1 accounting disable
ZXR10(config-nas)#aaa 1 multiple-hosts enable
ZXR10(config-nas)#aaa 1 default-isp zte163.net
ZXR10(config-nas)#aaa 1 fullaccount disable
ZXR10(config-nas)#aaa 1 authentication local
ZXR10(config-nas)#create localuser 1 name  A0001
ZXR10(config-nas)#localuser 1 mac 00d0.d0d0.1234
ZXR10(config-nas)#create localuser 2 name  A0002
ZXR10(config-nas)#localuser 2 mac 00d0.d0d0.1456
ZXR10(config-nas)#create localuser 3 name  A0003
ZXR10(config-nas)#localuser 3 mac 00d0.d0d0.1689
```

In the above configuration, local authentication function on the authenticator switch is enabled to implement the application requirement of the enterprise. According to the above configuration, only

00d0.d0d0.1234, 00d0.d0d0.1456 and 00d0.d0d0.1689 network card addresses are accessed and the Internet access duration of these three users, named as A0001, A0002 and A0003, is summed up. Duration is recorded on the Radius server.

# DOT1x Maintenance and Diagnosis

To configure Dot1x maintenance and diagnosis, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show dot1x** | This displays Dot1x authentication configuration information |
| 2 | ZXR10#**show aaa** [<*rule-id*>] | This displays an AAA control entry |
| 3 | ZXR10#**show aaa statistics** [<*rule-id*>] | This displays statistics information of rules |
| 4 | ZXR10#**show client** {**port** <*port-name*> **vlan** <*vlan-id*>\|**slot** <*slot-id*>{**aaa** <*rule-id*>\| **all** \| **index** <*id*>\| **mac** <*macaddr*>\| **vlan** <*vlanid*>}} | This displays online user information |
| 5 | ZXR10#**show client statistics** | This displays statistics information of online users |
| 6 | ZXR10#**show localuser** [<*user-id*>] | This displays information of local users |
| 7 | ZXR10#**debug nas** | This traces the transmitting and receiving packet and handling processes of the dot1x |
| 8 | ZXR10#**debug radius all** | This traces the process of interacting with the radius |

Chapter 12

# Cluster Management Configuration

**Table of Contents**

Cluster Management Overview ..... 121
Configuring Cluster Management ..... 123
Cluster Management Configuration Example..... 126
Cluster Management Maintenance and Diagnosis ..... 126

## Cluster Management Overview

Cluster is a combination of a group of switches in a specific broadcast domain. This group of switches forms a unified management domain which provides a public network IP address and a management interface to the outside and provides the functions of managing and accessing every member in the cluster.

Management switch is configured with public network IP address as a command switch and other managed switches such as member switches. Public network IP address is not configured for the member switch but a private address is assigned to the member switch with similar DHCP function of the command switch. Command switch and member switch form a cluster (private network).

It is recommended to isolate the broadcast domain of the public network and that of the private network on the command switch, and shield the direct access to the private address. The command switch provides a management and maintenance channel to the outside to manage the cluster in a centralized and unified manner.

A broadcast domain is composed of four kinds of switches:

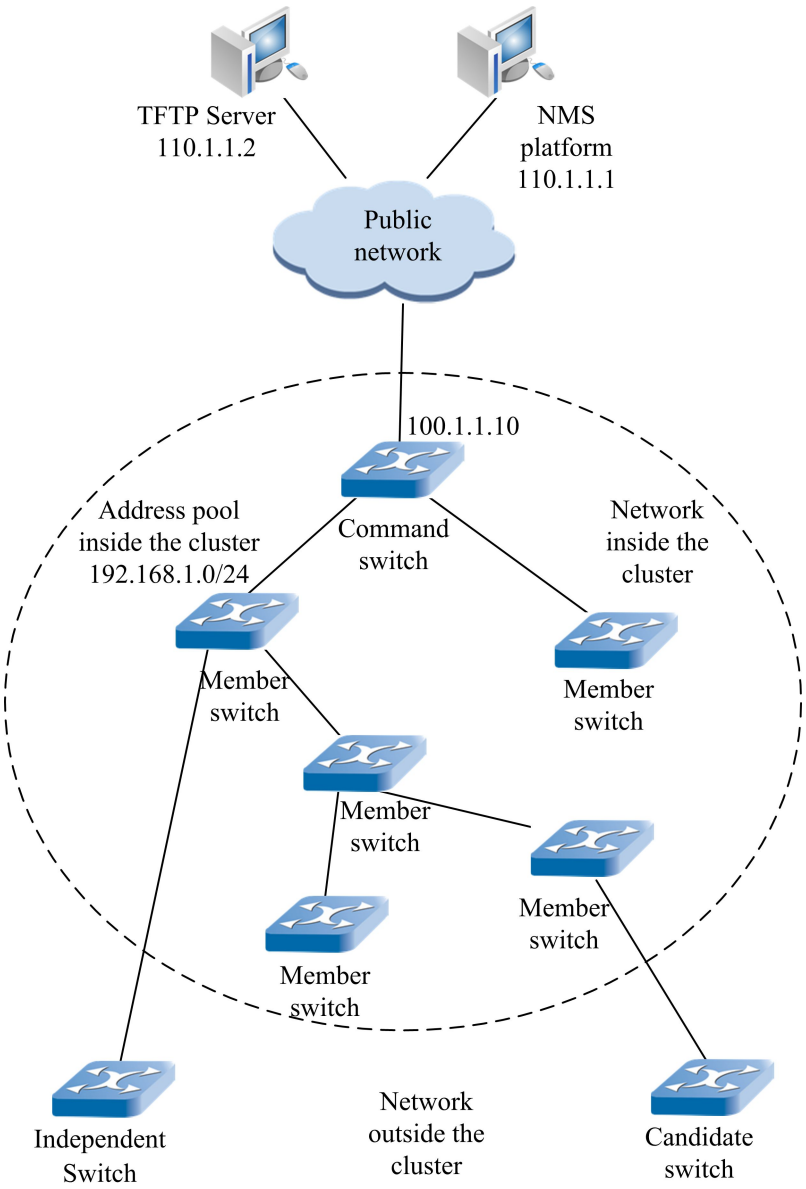
- Command switch
- Member switch
- Candidate switch
- Independent switch

There is only one command switch in a cluster. Command switch can collect equipment topology and establish a cluster automatically. After the cluster is established, command switch provides a management channel for cluster to manage member switch. Mem-

ber switch serves as a candidate switch before being added into cluster. Switch which does not support member switch is called independent switch.

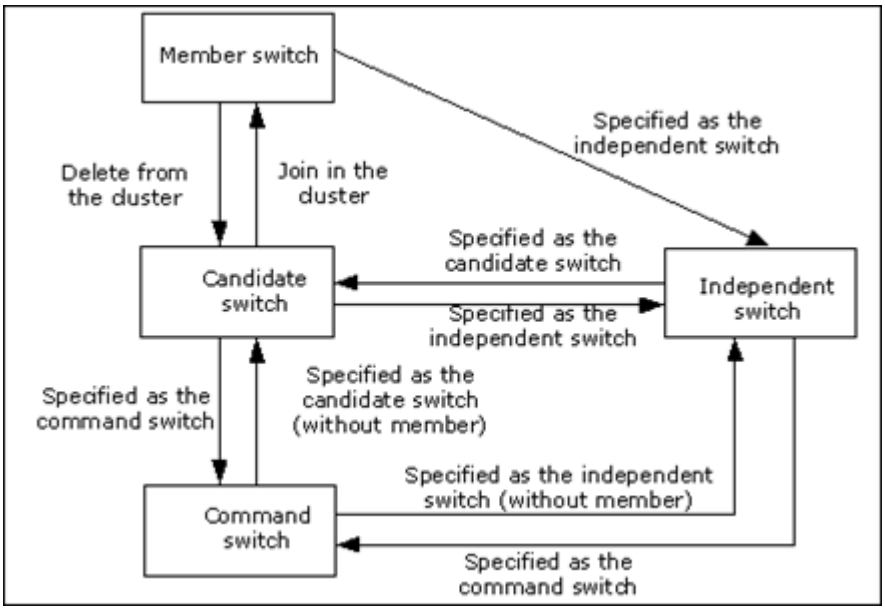
Cluster management network is formed as shown in Figure 32.

**FIGURE 32 CLUSTER MANAGEMENT NETWORK**




Switching rule of four kinds of switches in the cluster is shown in Figure 33.

**FIGURE 33 SWITCHING RULE**


Member switch
Specified as the independent switch
Delete from the duster
Join in the duster
Specified as the candidate switch
Candidate switch
Independent switch
Specified as the independent switch
Specified as the candidate switch (without member)
Specified as the command switch
Specified as the independent switch (without member)
Command switch
Specified as the command switch

# Configuring Cluster Management

## Enabling ZDP

To enable ZTE Discovery Protocol (ZDP), perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**zdp enable** | This enable ZDP function globally |
| 2 | `ZXR10(config)#`**interface** <*interface-name*> | This enters interface configuration mode |
| 3 | `ZXR10(config-if)#`**zdp enable** | This enable ZDP function on an interface |
| 4 | `ZXR10(config-if)#`**exit** | This exits interface configuration mode |
| 5 | `ZXR10(config)#`**zdp timer** <*time*> | This configures time interval of transmitting ZDP packets |
| 6 | `ZXR10(config)#`**zdp holdtime** <*time*> | This configures valid holding time of ZDP information |

# Enabling ZTP

To enable ZTE Topology Protocol (ZTP), perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**ztp enable** | This enables ZTP function globally |
| 2 | ZXR10(config)#**interface** <*interface-name*> | This enters interface configuration mode |
| 3 | ZXR10(config-if)#**ztp enable** | This enables ZTP function on an interface |
| 4 | ZXR10(config-if)#**exit** | This exits interface configuration mode |
| 5 | ZXR10(config)#**ztp vlan** <*vlanID*> | This conducts ZTP topology collection on different VLANs |
| 6 | ZXR10(config)#**ztp hop** <*number*> | This sets the number of hops of ZTP topology collection |
| 7 | ZXR10(config)#**ztp hop-delay** <*time*> | This sets each hop delay in sending ZTP protocol packets |
| 8 | ZXR10(config)#**ztp port-delay** <*time*> | This sets delay in sending ZTP protocol packets on the port |
| 9 | ZXR10(config)#**ztp start** | This conducts once topology collection |
| 10 | ZXR10(config)#**ztp timer** <*time*> | This sets ZTP timing topology collection time |

# Setting up a Cluster

To set up a cluster, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**group switch-type** { **candidate** \| **independent** \|{ **commander** [ **iip-pooll** < *ip_addr*>{ **maassk** < *net-mask*>\| **llengtth** < *mask_len*>}]}} | This configures the role of a switch and assigns an IP address pool to the cluster. |
| 2 | ZXR10(config)#**group name** <*name*> | This changes the name of a cluster. |
| 3 | ZXR10(config)#**group handtime** <*time*> | This configures the handshake time. |
| 4 | ZXR10(config)#**group holdtime** <*time*> | This configures holdtime between member switch and command switch on a commander switch. |

| Step | Command | Function |
|---|---|---|
| 5 | ZXR10(config)#**group time synchronize** | This enables clock synchronization for cluster management. |
| 6 | ZXR10(config)#**group member** { **all-candidates** \| **deviice** < *device-id*>\|{ **maac** < *mac-address*>[ **memberr** < *member-id*>]}} | This adds a designated device or MAC address as a member on a commander switch. |

# Maintaining a Cluster

To maintain a cluster, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**group reset-member** {**all** \|<*member_id*>} | This restart the member on the command switch |
| 2 | ZXR10(config)#**group save-member** {**all** \|<*member_id*>} | This saves the member configuration on the command switch |
| 3 | ZXR10(config)#**group erase-member** {**all** \|<*member_id*>} | This deletes the member configuration file from the command switch |
| 4 | ZXR10(config)#**group tftp-server** <*ip_addr*> | This configures the tftp server on the cluster |
| 5 | ZXR10(config)#**group trap-host** <*ip_addr*> | This configures the alarm receiver of the cluster |

# Configuring Cluster Operation Commands

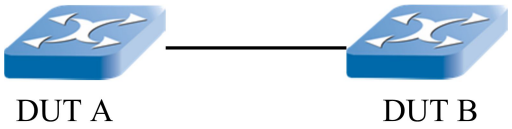
To configure cluster operation commands, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**rlogin** | This logs in from the command switch to member switch or from the member switch to command switch |
| 2 | ZXR10#**copy** <*source-device*><*source-file*><*destination-device*><*destination-file*> | This uploads or downloads files through the cluster tftp server on the member switch |

# Cluster Management Configuration Example

This example describes how to connect two devices to implement cluster management, as shown in Figure 34.

**FIGURE 34 CLUSTER MANAGEMENT CONFIGURATION EXAMPLE**




Configuration steps are as follows:

1. Ensure that two ports are in a VLAN (configured as vlan1 and ensure that vlan1 does not configure Layer 3 address).
2. Execute **show zdp neighbor** on DUT A and ensure zdp neighbor is already set up.
3. Execute **ztp start** on DUT A to conduct topology collection, and then execute **show ztp device-list** to view DUT A and DUT B.
4. Configure DUT A as command switch with **group switch-type** command. View command switch with **show group** command.
5. Configure DUT B as the member switch with **group member device 1** command and then view Member 1 in the up state with the **show group member** command.
6. Log in to Member 1 with the **rlogin member 1** command in the privilege mode, and log in from Member 1 to the command switch with the **rlogin commander** command.

# Cluster Management Maintenance and Diagnosis

To configure cluster management maintenance and diagnosis, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show zdp** | This displays ZDP configuration information |
| 2 | ZXR10#**show ztp** | This displays ZTP configuration information |
| 3 | ZXR10#**show group** | This displays cluster configuration information |
| 4 | ZXR10#**show zdp neighbour** [{**interface** *<interface>*}\|{**mac** *<mac id>*}] | This displays ZDP neighbor |

| Step | Command | Function |
|---|---|---|
| 5 | ZXR10#**how zdp device-list** | This displays received equipment information |
| 6 | ZXR10#**show group member** [**member-num** <*mem_id*>] | This displays group member information |

**Note:**

To trace transmitting and receiving packets condition and handling condition of cluster management processes ZDP and ZTP with **debug group** command.

This page is intentionally blank.

Chapter 13

# Network Management Configuration

**Table of Contents**

NTP Configuration ..... 129
RADIUS Configuration ..... 130
SNMP Configuration ..... 133
RMON Configuration ..... 134
SysLog Configuration ..... 136
LLDP Configuration ..... 138

# NTP Configuration

## NTP Overview

Network Time Protocol (NTP) is the protocol used to synchronize the clocks of computers on a network or across multiple networks, like the Internet. Without adequate NTP synchronization, organizations cannot expect their network and applications to function properly. ZXR10 8900 series switch acts as the NTP client.

## Configuring NTP

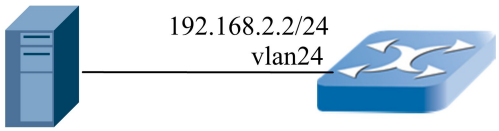
To configure NTP, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**ntp server** <*ip-address*>[**version** <*number*>] | This defines a time server |
| 2 | `ZXR10(config)#`**ntp enable** | This enables NTP function |
| 3 | `ZXR10(config)#`**ntp source** <*ip-address*> | This configures the source address |
| 4 | `ZXR10(config)#`**show ntp status** | This displays NTP running state |

## NTP Configuration Example

This example shows routing switch as an NTP client and assume that the NTP protocol version is 2. Network topology is shown in Figure 35.

**FIGURE 35 NTP CONFIGURATION EXAMPLE**




192.168.2.1/24
NTP server

ZXR10 configuration:

```
ZXR10(config)#interface vlan24
ZXR10(config-if)#ip address 192.168.2.2 255.255.255.0
ZXR10(config-if)#exit
ZXR10(config)#ntp enable
ZXR10(config)#ntp server 192.168.2.1 version 2
```

# RADIUS Configuration

## Radius Overview

Remote Authentication Dial In User Service (RADIUS) is a standard AAA protocol. AAA represents Authorization, Authentication and Accounting. AAA is used to authenticate users accessing the routing switch and prevent accessing of illegal users, thus enhancing security of the equipment. What's more, services like DOT1X can also use RADIUS server for authentication and accounting.

ZXR10 8900 series switch supports RADIUS authentication function to authenticate Telnet users accessing routing switch.

ZXR10 8900 series switch supports multiple RADIUS server groups. Four authentication servers can be configured in each RADIUS group. Server timeout time and max retry times for timeout can be set for each group. Administrator can configure different RADIUS groups to select a specific RADIUS server.

## Configuring a RADIUS Accounting Group

To configure RADIUS accounting group, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**radius accounting-group** <*group-number*> | This configures RADIUS accounting group |

# Configuring a RADIUS Authentication Group

To configure RADIUS authentication group, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**radius authentication-group** <*group-number*> | This configures RADIUS authentication group |

# Configuring RADIUS Parameters

To configure RADIUS parameters, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config-acctgrp-1)#**timeout** <*timeout*> | This configures RADIUS timeout |
| 2 | ZXR10(config-acctgrp-1)#**algorithm** {**first** \| **round-robin**} | This configures algorithm of RADIUS server |
| 3 | ZXR10(config-acctgrp-1)#**alias** <*name-str*> | This configures byname of RADIUS server group |
| 4 | ZXR10(config-acctgrp-1)#**calling-station-format** <*Format number*> | This defines format of calling-station-id field |
| 5 | ZXR10(config-acctgrp-1)#**deadtime** <*time*> | This configures dead-time of authentication server |
| 6 | ZXR10(config-acctgrp-1)#**local-buffer** {**enable** \| **disable**} | This clears local buffer of accounting server |
| 7 | ZXR10(config-acctgrp-1)#**max-retries** <*times*> | This configures retransmission times of RADIUS server |
| 8 | ZXR10(config-acctgrp-1)#**nas-ip-address** <*NAS IP address*> | This configures nas-ip of RADIUS server |
| 9 | ZXR10(config-acctgrp-1)#**server** <*number*><*ipaddress*> **key** <*keystr*> **port** <*portnum*> | This configures RADIUS server and its parameters |

| Step | Command | Function |
|---|---|---|
| 10 | ZXR10(config-acctgrp-1)#**user-name-format** {**include-domain** \| **strip-domain**} | This configures format of name sent to RADIUS server by BRAS |
| 11 | ZXR10(config-acctgrp-1)#**vendor** {**enable** \| **disable**} | This enables or disables attributes defined by vendor in RADIUS protocol packets |

# Viewing RADIUS Information

To view RADIUS information, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show counter radius all** | This displays statistics information |
| 2 | ZXR10#**show accounting local-buffer all** | This displays all information in local buffer |
| 3 | ZXR10#**debug radius all** | This displays RADIUS debugging information |

**Note:**

To clear all information in local buffer, use **clear accounting loca l-buffer all** command.

# RADIUS Configuration Example

This example describes how to configure a RADIUS accounting group. Procedure of configuring a RADIUS authentication group is the same.

```
ZXR10(config)#radius accounting-group 1
ZXR10(config-acct-group-1)#algorithm round-robin
ZXR10(config-acct-group-1)#calling-station-format 2
ZXR10(config-acct-group-1)#deadtime 5
ZXR10(config-acct-group-1)#local-buffer enable
ZXR10(config-acct-group-1)#max-retries 5
ZXR10(config-acct-group-1)#nas-ip-address 10.1.1.4
ZXR10(config-acct-group-1)#server 1 10.2.1.3 key uas
ZXR10(config-acct-group-1)#server 2 12.1.2.3 key uas
ZXR10(config-acct-group-1)#timeout 10
```

# SNMP Configuration

## SNMP Overview

SNMP is one of the most popular network management protocols. This protocol enables a network management server to manage all the devices in a network.

SNMP is managed based on server and client. Background NMS server serves as SNMP server and foreground network device serves as SNMP client. Foreground and background share an MIB and communicate with each other through SNMP protocol. It is required to configure specific SNMP server for the rouging switch as SNMP agent and define contents and authorities availably collected by NMS. ZXR10 8900 series switch supports multiple versions of SNMP.

## Configuring SNMP

SNMPv1/v2c adopts the community authentication mode. SNMP community is named by strings and different communities have read-only or read-write access authorities. Community with read-only authority can only query equipment information. Community with read-write authority can configure the equipment.

Both read-only and read-write are limited by the view. Operations can only be conducted in the permitted view range. When parameter view is omitted use default view and use parameter ro if ro/rw are omitted.

To configure SNMP, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**snmp-server community** <*community-name*>[**view** <*view-name*>][**ro**\|**rw**] | This sets community name in an SNMP message |
| 2 | ZXR10(config)#**snmp-server view** <*view-name*><*subtree-id*>{**included**\|**excluded**} | This defines an SNMPv2 view |
| 3 | ZXR10(config)#**snmp-server contact** <*mib-sysconta ct-text*> | This sets system contact for an MIB object |
| 4 | ZXR10(config)#**snmp-server location** <*mib-syslocati on-text*> | This sets the type of trap allowed to be sent by a proxy |
| 5 | ZXR10(config)#**snmp-server enable trap** [<*notification-type*>] | This configures trap type |
| 6 | ZXR10(config)#**snmp-server host** {{<*ip-address*>{**i nform** \| **trap**} **version** {**1** \| **2c** \| **3**}<*community*>}\| **mng** \| **vrf**} | This configures the sending address, port, version and inform for the host |

| Step | Command | Function |
|---|---|---|
| 7 | `ZXR10(config)#`**show snmp** | This displays the statistics on SNMP messages |
| 8 | `ZXR10(config)#`**show snmp config** | This displays configuration information of SNMP module |

**Note:**

- For step 2, **include** or **exclude** adds or removes <*subtree-ID*> from specified view. Configurations are allowed for many times for the same <*view-name*>, which results in a set of cooperating commands.
- For step 3, sysContact is a management variable in system group in MIB II. It contains ID and contact of the person relevant to a managed device.
- For step 4, sysLocation is a management variable in system group in MIB II. It contains the positions of managed devices.
- For step 5, Trap is the information a managed device sends to Network Management System (NMS) without request. It is used to report emergent and important events.
- For step 6, ZXR10 8900 series switch supports 5 types of conventional traps: snmp, bgp, ospf, rmon and stalarm.

## SNMP Configuration Example

This example describes the configuration of SNMP.

```
ZXR10(config)#snmp-server view myViewName 1.3.6.1.2.1 included
ZXR10(config)#snmp-server community myCommunity view myview rw
ZXR10(config)#snmp host 168.1.1.1 ver 1 community-name ospf
ZXR10(config)#snmp-server location  this is ZXR10 in china
ZXR10(config)#snmp-server contant this is ZXR10, tel: (025)2872006
```

# RMON Configuration

## RMON Overview

Remote Monitoring (RMON) system is to monitor network terminal services. A remote detector, that is the routing switch system, completes data collection and processing through RMON. Routing switch contains RMON agent software communicating with the NMS through the SNMP. Information is usually transmitted from the routing switch to the NMS when necessary.

# Configuring RMON

To configure RMON, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config-if)#**rmon collection statistics** <*index*>[**owner** <*string*>] | This enables statistics on a port |
| 2 | ZXR10(config-if)#**rmon alarm** <*index*><*variable*><*interval*>{**delta**\|**absolute**} **rising-threshold** <*value*>[<*event-index*>] **falling-threshold** <*value*>[<*event-index*>][**owner** <*string*>] | This sets alarms and MIB objects |
| 3 | ZXR10(config-if)#**rmon collection history** <*index*>[**owner** <*string*>][**buckets** <*bucket-number*>][**interval** <*seconds*>] | This enables history collection of the interface |
| 4 | ZXR10(config-if)#**rmon event** <*index*>[**log**][**trap** <*community*>][**description** <*string*>][**owner** <*string*>] | This configures an event |
| 5 | ZXR10(config-if)#**show rmon** [**alarms**][**events**][**history**][**statistics**] | This displays RMON configuration and related information |

# RMON Configuration Example

The following are several configuration examples of the RMON.

**Example** This example shows how to configure and start statistics control entries of the RMON.

```
ZXR10(config)#interface fei_1/1
ZXR10(config-if)#rmon collection statistics 1 owner rmontest
```

Assume n computers are linked to port fei_1/1 and when these computers communicate on the sub-network, traffic statistics can be viewed through NMS software and it can also be viewed with **show** command.

```
ZXR10#show rmon statistics
EtherStatsEntry 1 is active, and owned by rmontest
Monitors ifEntry.1.1 which has
Received 60739740 octets, 201157 packets,
1721 broadcast and 9185 multicast packets,
0 undersized and 0 oversized packets,
0 fragments and 0 jabbers,
0 CRC alignment errors and 32 collisions.
# of dropped packet events (due to lack of resources): 511
# of packets received of length (in octets):
64: 92955, 65-127: 14204, 128-255: 1116,
256-511: 4479, 512-1023: 85856, 1024-1518:2547
```

**Example** This example describes how to configure and enable RMON history control entry.

```
ZXR10(config)#interface fei_1/1
ZXR10(config-if)#rmon collection history 1 bucket 10
interval 10 owner rmontest
```

Use **show** command to view the RMON history information.

```
ZXR10#show rmon history
Entry 1 is active, and owned by rmontest
Monitors ifEntry.1.1 every 10 seconds
Requested # of time intervals, ie buckets, is 10
Granted # of time intervals, ie buckets, is 10
Sample # 1 began measuring at 00:11:00
Received 38346 octets, 216 packets,
0 broadcast and 80 multicast packets,
0 undersized and 0 oversized packets,
0 fragments and 0 jabbers,
0 CRC alignment errors and 0 collisions.
# of dropped packet events is 0
Network utilization is estimated at 1
```

**Example** This example describes how to configure and enable RMON alarm control entry.

```
ZXR10(config)#rmon alarm 1 system.3.0 10 absolute
rising-threshold 1000 1 Falling-threshold 10 0 owner rmontest
```

Use **show** command to view RMON alarm information.

```
ZXR10#show rmon alarm
Alarm 1 is active, owned by rmontest
Monitors system.3.0 every 10 seconds
Taking absolute samples, last value was 54000
Rising threshold is 1000, assigned to event 1
Falling threshold is 10, assigned to event 0
On startup enable rising or falling alarm
```

**Example** This example describes how to configure and enable event.

```
ZXR10(config)#rmon event 1 log trap rmontrap description test owner rmontest
```

After configuring an alarm control entry and wait for 10s, use **s how** command to view the contents of the RMON event.

```
ZXR10#show rmon event
Event 1 is active, owned by rmontest
Description is test
Event firing causes log and trap to community rmontrap,
last fired 05:40:20
Current log entries:
        index       time   description
         1        05:40:14   test
```

# SysLog Configuration

## SysLog Overview

ZXR10 8900 series switch allows user to set and query logs. Log information makes it easy for maintaining routing switch regularly. Log information allows viewing alarm information and port status changes on routing switch. Logs can be displayed on the configured terminals in real time, or saved on routing switch or a background log server in files. It can enable SysLog protocol on ZXR10 8900 series switch to transmit logs by communicating with background syslog server through the protocol.

# Configuring SysLog

To configure SysLog, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**logging on** | This enables log |
| 2 | ZXR10(config)#**logging buffer** <*buffer-size*> | This set log buffer size |
| 3 | ZXR10(config)#**logging mode** <*mode*>[<*interval*>] | This sets a log cleanup mode |
| 4 | ZXR10(config)#**logging console** <*level*> | This sets level of logs to be displayed on a console interface or telnet interface |
| 5 | ZXR10(config)#**logging level** <*level*> | This sets the level of logs to be saved in the log cache |
| 6 | ZXR10(config)#**logging ftp** <*level*>[**vrf** <*vrf-name*>\|**mng**]<*ftp-server*><*username*><*password*>[<*filename*>] | This sets the parameters of FTP log server |
| 7 | ZXR10(config)#**syslog on** | This enables SysLog protocol processing |
| 8 | ZXR10(config)#**syslog level** <*level*> | This sets a log level for SysLog protocol processing |
| 9 | ZXR10(config)#**syslog server** [**vrf** <*vrf-name*>\|**mng**]<*ip-address*>[**fport** <*fport*>][**lport** <*lport*>] | This sets the parameters of the background SysLog server |
| 10 | ZXR10(config)#**show logging alarm** {[**typeid** <*type*>][**start-date** <*date*>][**end-date** <*date*>][**level** <*level*>]} | This displays log information |

**Note:**

In step 10, types of supported alarmed information include environment, board, port, ROS, database, OAM, security, OSPF, RIP, BGP, DRP, TCP-UDP, IP, IGMP, Telnet, ARP, ISIS, ICMP, SNMP and RMON.

# SysLog Configuration Example

This example describes the setting SysLog. Before configuring SysLog, enable the log function with **logging on** command.

```
ZXR10(config)#logging on
ZXR10(config)#logging buffer 100
ZXR10(config)#logging mode FULLCLEAR
ZXR10(config)#logging console warnings
ZXR10(config)#logging level errors
```

# LLDP Configuration

## LLDP Overview

Link Layer Discovery Protocol (LLDP) is a new protocol defined in 802.1ab. It enables that neighbor devices can send messages to each other. LLDP is used to update physical topology information and create a device management information database.

**Working Flow**

The working flow of LLDP is described as follows:

1. Local device sends link and management information to neighbor devices.
2. Local device receives network management information from neighbor devices.
3. Local device saves network management information received from neighbor devices in MIB. Network management software can search the connection information of link layer in the MIB.

**Function**

LLDP is neither a configuration protocol of remote systems, nor a signal control protocol for ports. LLDP only finds out the difference of Layer 2 protocol configuration on neighbor devices and reports the problem to upper layer. It does not provide corresponding mechanism to solve the problems.

Generally speaking, LLDP is a kind of neighbor discovery protocol, providing a standard for devices in Ethernet, such as switches, routers and wireless LAN access points. It helps the devices to tell the neighbors its existence and saves discovery information of the neighbors. Information such as configuration and device identifier can be notified by LLDP.

**LLDPDU**

LLDP defines a universal advertisement set, a protocol for notifying advertisement messages and a method to save received advertisement messages. The devices can use a Link Layer Discovery Protocol Data Unit (LLDPDU) to notify multiple advertisement messages.

**TLV**

The LLDPDU contains a short message unit of a variable length, called Type Length Value (TLV).

- Type: the type of the message to be sent
- Length: the byte number of the message to be sent
- Value: the effective information of the message to be sent

Each LLDPDU includes four compulsory TLVs and an optional TLV:

- Device ID TLV
- Port ID TLV
- TTL TLV
- Optional TLV
- LLDPUD ending TLV

Device ID TLV and port ID TLV are used to identify the senders.

TTL TLV tells the receivers the hold time of the message. If the receiver does not receive update information from the sender within the hold time, the receiver will discard all related messages. IEEE

has defined a recommendatory update frequency, that is, the update messages should be sent every 30 seconds.

Optional TLV contains a basic management TLV set, an IEEE 802.1-organized particular TVL, and an IEEE 802.3-organized particular TVL.

The appearance of LLDPUD ending TLV means the end of the LLDPDU.

# Configuring LLDP

To configure LLDP, perform the following steps.

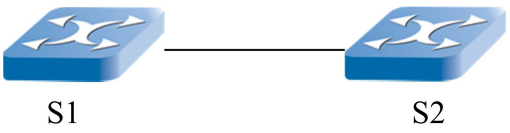
| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**lldp enable** | This enables LLDP. |
| 2 | `ZXR10(config)#`**lldp hellotime** <*seconds*> | This configures the interval of sending LLDPDUs. |
| 3 | `ZXR10(config)#`**lldp holdtime** <*multiple*> | This configures the aging time of LLDPDU. The product of parameters multiple and hellotime is aging time. |
| 4 | `ZXR10(config)#`**interface** < *interface-name*> | This enters interface configuration mode. |
| 5 | `ZXR10(config-if)#`**lldp setAdminStatus** {**enabledtxrx** \| **rxonly** \| **txonly**\| **disabled**} | This configures the management state of LLDP. |

# LLDP Configuration Example

This example shows how to configure LLDP.

As shown in Figure 36, S1 connects to S2. Configure LLDP on the two switches to make them discover each other.

**FIGURE 36 LLDP CONFIGURATION EXAMPLE**




Configuration of S1:

```
Zxr10#conf t
Zxr10(config)#lldp enable interface gei_1/1
```

Configuration of S2:

```
Zxr10#conf t
Zxr10(config)#lldp enable interface gei_1/1
```

Show configuration results:

- Showing global information of line card

```
Zxr10#show lldp config
-------------------------------------
Lldp enable: enabledRxTx
Lldp hellotime: 30s
Lldp holdtime: 120s
Lldp maxneighbor: 128
Lldp curneighbor: 28
-------------------------------------
```

- Showing interface information

```
Zxr10#show lldp config interface gei_1/1
Lldp port enable: enabledRxTx
Lldp maxneighbor: 8
Lldp curneighbor: 0
-------------------------------------
```

- Showing neighbor information of line card

```
Zxr10#show lldp neighbor
Capability Codes: R - Router, T - Trans Bridge, B - Source
Route Bridge, S - Switch, H - Host, I - IGMP, r - Repeater,
P - Phone W - W
LAN Access Point
Local Intrfce Device ID Holdtime Capability Platform Port ID
------------------------------------------------------------
gei_1/3 00d0d0c7ffe0 120 B S ZXR10 ROS Version gei_1/2
V4.08.23 ZX..
gei_1/2 00d0d0c7ffe0 120 B S ZXR10 ROS Version gei_1/3
V4.08.23 ZX..
gei_1/5 00d0d0c7ffe0 120 B S ZXR10 ROS Version gei_1/
```

- Showing interface neighbor information

```
Zxr10#show lldp neighbor interface gei_1/1
c Capability Codes: R - Router, T - Trans Bridge,
B - Source Route Bridge, S - Switch, H - Host, I - IGMP,
r - Repeater, P - Phone W - W
LAN Access Point
Local Intrfce Device ID Holdtime Capability Platform Port ID
------------------------------------------------------------
gei_1/1 0019c6059fc0 99 B S ZXR10 ROS Version gei_1/1V4.08.23 ZX..
```

Chapter 14

# IPTV Configuration

**Table of Contents**

IPTV Overview ..... 141
Configuring IPTV ..... 141
IPTV Configuration Example ..... 145
IPTV Maintenance and Diagnosis ..... 146

# IPTV Overview

Internet Protocol Television (IPTV) is also called Interactive Network TV. IPTV is a method of distributing television content over IP that enables a more customized and interactive user experience. IPTV allows people who are separated geographically to watch a movie together, while chatting and exchanging files simultaneously. IPTV uses a two-way broadcast signal that is sent through the service provider's backbone network and servers. It allows the viewers to select content on demand, and take advantage of other interactive TV options. IPTV can be used through PC or "IP machine box + TV".

# Configuring IPTV

## Configuring IPTV Global Parameters

To configure IPTV global parameters, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**iptv control** {**enable**\|**disable**} | This configures IPTV function |
| 2 | `ZXR10(config)#`**iptv cac** {**enable** \| **disable**} | This configures IPTC Channel Access Control (CAC) function |

| Step | Command | Function |
|---|---|---|
| 3 | ZXR10(config)#**iptv sms-server** <*server-ip*> | This configures the IP address of service management system server |
| 4 | ZXR10(config)#**iptv sms-server-port** <*port-number*> | This configures the port of service management system server |

# Configuring Global Parameters of IPTV Preview

To configure global parameters of IPTV preview, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**iptv prw** {**enable** \| **disable**} | This configures IPTV preview function |
| 2 | ZXR10(config)#**iptv prw reset** | This resets preview function |
| 3 | ZXR10(config)#**iptv prw auto-reset-time** <*HH:MM:SS*> | This configures the auto-reset time of preview |
| 4 | ZXR10(config)#**iptv prw recognition-time** <*recog-time*> | This configures recognition time of preview |
| 5 | ZXR10(config)#**iptv prw overcout-cdr** {**enable** \| **disable**} | This configures whether to generate CDR record when maximum preview times are over |

# Configuring IPTV CDR Parameters

To configure CDR parameters, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**iptv cdr** {**enable**\|**disable**} | This configures CDR function |
| 2 | ZXR10(config)#**iptv cdr max-records** <*cdr-size*> | This sets the maximum number of CDR record |
| 3 | ZXR10(config)#**iptv cdr report** | This reports CDR manually |
| 4 | ZXR10(config)#**iptv cdr report-interval** <*report-interval*> | This configures the interval to report CDR |

| Step | Command | Function |
|---|---|---|
| 5 | ZXR10(config)#**iptv cdr create-period** <*period*> | This configures the cycle to generate CDR for allowing users to watch programs for long time |
| 6 | ZXR10(config)#**iptv cdr deny-right** {**enable**\|**disable**} | This configures whether to generate CDR when access privilege is configured deny |
| 7 | ZXR10(config)#**iptv cdr prw-right** {**enable**\|**disable**} | This configures whether to generate CDR when access privilege is configured preview |
| 8 | ZXR10(config)#**iptv cdr warning-threshold** <*threshold value*> | This configures the alarm threshold value of CDR cache pool |
| 9 | ZXR10(config)#**iptv cdr report-threshold** <*threshold value*> | This configures the threshold value to send CDR |

# Configuring IPTV Channels

To configure IPTV channels, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**iptv channel mvlan** < *vlan-id*> **group** < *group-ip*>[{ **name** < *channel-name* >[ **id** < *channel-id*>]}\|{ **count** < *count-value*>[ **prename** < *prename-str*>]}] | This creates channels of IPTV. |
| 2 | ZXR10(config)#**iptv channel name** < *old-name*> **rename**< *new-name*> | This sets the name of a channel. |
| 3 | ZXR10(config)#**iptv channel** { **name** \| **idlist**}< *channel-name*>{ **viewfile-name** < *viewfile-name*>\| **viewfile-id**< *viewfile-id*>} | This configures a preview configuration file for a channel. |
| 4 | ZXR10(config)#**iptv channel** { **idlist** \| **name**}< *channel-idlist*> **cdr** { **enable** \| **disable**} | This configures whether to enable logging function for a channel. |
| 5 | ZXR10(config)#**no iptv channel** {**idlist**< *channel-idlist*>\| **all** \| **name** < *channel-name*>} | This deletes channels. |

# Configuring IPTV Service Package

To configure IPTV service package, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**iptv package name** <*package-name* >[**pkgid** <*package-id*>] | This creates an IPTV service package |

| Step | Command | Function |
|---|---|---|
| 2 | ZXR10(config)#**iptv package** <*package-name*> **channel** < *idlist*>{**deny**\|**permit**\|**preview**} | This adds a channel to the package and sets the privilege of the channel |
| 3 | ZXR10(config)#**no iptv package** {**all** \|{ **package-name** [<*package-name*>]\| **package-id** [<*package-id*>]} **channel idlist**>} | This deletes the package or a channel in the package |

**Note:**

Package ID and name are unique. When package ID is not configured, the system assigns an ID for the package automatically.

# Configuring IPTV Preview Template

To configure IPTV preview template, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**iptv view-profile name** <*viewfile-name*>[ **id** < *viewfile -id*>] | This creates a preview configuration file |
| 2 | ZXR10(config)#**iptv view-profile name** <*viewfile-name*> **count** <*view-count*> | This configures the maximum preview times |
| 3 | ZXR10(config)#**iptv view-profile name** <*viewfile-name*> **duration** <*view-duration*> | This configures the maximum duration for single preview |
| 4 | ZXR10(config)#**iptv view-profile name** <*viewfile-name*> **blackout** <*view-interval*> | This configures the minimum preview interval |
| 5 | ZXR10(config)#**no iptv view-profile** { **all** \| **viewfile-name** < *viewfile-name* >\| **viewfile-id** < *viewfile-id* >} | This deletes the preview template |

# Configuring CAC

To configure Channel Access Control (CAC), perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**interface** < *interface-name*> | This enters interface configuration mode. |
| 2 | ZXR10(config-if)#**iptv** [ **vlan** {<*vlan-idlist*>\|<*vlan-name*>}] **service** { **start** \| **pause** \| **resume** \| **remove**} | This configures current service state of user. |

| Step | Command | Function |
|---|---|---|
| 3 | ZXR10(config-if)#**iptv** [**vlan**{<*vlan-id*>\|<*vlan-name*>}] **control-mode** {**package** \| **channel**} | This configures multicast control mode for user. |
| 4 | ZXR10(config-if)#**iptv** [**vlan** {<*vlan-idlist*>\|<*vlan-name*>}] **package** {**name** <*package-name*>\| **idlist** <*package-idlist*>} | This assigns package for user. |
| 5 | ZXR10(config-if)#**iptv** [**vlan** {<*vlan-idlist*>\|<*vlan-name*>}] **channel**{**name** <*channel-name*>\| **idlist** <*channel-idlist*>}{**deny**\|**permit**\|**preview**\|**query**} | This configures the channel access privilege of user interface. |
| 6 | ZXR10(config-if)#**iptv** [**vlan** {<*vlan-idlist*>\|<*vlan-name*>}] **cdr** {**enable** \| **disable**} | This configures whether to generate CDR record. |
| 7 | ZXR10(config-if)#**iptv** [ **vlan** {< *vlan-idlist*>\|< *vlan-name*>}] **max-access** < *channel-num*> | This sets max user accesses to channel. |
| 8 | ZXR10(config-if)#**no iptv** [{ **vlan-id** < *vlan-id*>\| **vlan-name** < *vlan-name*>}] **package**{ **name** < *package-name*>\| **idlist** < *package-idlist*>} | This deletes package allocated to rule. |

## Configuring IPTV Fast Leave

To configure IPTV fast leave, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config)#**iptv fast-leave mvlan** < *mvlan-id*> | This enables IPTV fast leave function. To enable this function, igmp snooping function must be enabled in mvlan. |
| 2 | ZXR10(config)#**no iptv fast-leave mvlan** < *mvlan-id*> | This disables IPTV CAC. |

## Managing IPTV Users

To manage IPTV users, use the following command.

| Command | Function |
|---|---|
| ZXR10(config)#**clear iptv client** [{{**slot** <*slot-number*> **index** <*client-index*>}\| **port** <*port-name*>\| **vlan** <*vlan-id*>}] | This manages IPTV users |

# IPTV Configuration Example

**Example** User who connects to port gei_1/1 is a requesting user of multicast group 224.1.1.1. Vlan ID of this multicast group is 100. There is only one channel with ID of 0. Configuration is shown below.

```
ZXR10(config)#iptv control enable
ZXR10(config)#iptv cac enable
ZXR10(config)#iptv channel mvlan 100 group 224.1.1.1
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#iptv service start
ZXR10(config-if)#iptv control-mode channel
ZXR10(config-if)#iptv channel id 0
```

**Example** User who connects to port gei_1/1 in Vlan1 is the preview user of multicast group 224.1.1.1. Max preview time is 2 minutes. Least preview interval is for 20 seconds. Max preview counts are 10. Vlan ID of multicast group is 100. There is only one channel with ID of 0. Configuration is shown below.

```
ZXR10(config)#iptv control enable
ZXR10(config)#iptv cac enable
ZXR10(config)#iptv channel mvlan 100 group 224.1.1.1
ZXR10(config)#iptv view-profile name vw1
ZXR10(config)#iptv view-profile name vw1 duration 120
ZXR10(config)#iptv view-profile name vw1 blackout 20
ZXR10(config)#iptv view-profile name vw1 count 10
ZXR10(config)#iptv channel id-list 0 viewfile-name vw1
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#iptv vlan 1 service start
ZXR10(config-if)#iptv vlan 1 control channel
ZXR10(config-if)#iptv vlan 1 channel id 0
```

**Example** Port gei_1/1 only allows receiving the querying packets of multicast group 224.1.1.1. Vlan ID of this multicast group is 100. There is only one channel with ID of 0. Configuration is shown below.

```
ZXR10(config)#iptv control enable
ZXR10(config)#iptv cac enable
ZXR10(config)#iptv channel mvlan 100 group 224.1.1.1
ZXR10(config)#interface gei_1/1
ZXR10(config-if)#iptv vlan 100 channel id 0 query
```

# IPTV Maintenance and Diagnosis

To locate IPTV problems and perform troubleshooting, execute related debugging commands. Here some **show** commands are introduced.

| Command | Function |
|---|---|
| ZXR10#**show iptv control** | This shows global configuration of IPTV. |
| ZXR10#**show iptv prw** | This shows global parameter configuration of IPTV preview. |
| ZXR10#**show iptv cdr** | This shows CDR configuration information. |
| ZXR10#**show iptv cdr record idlist** <*cdr-idlist*> | This shows information of generated CDR records. |

| Command | Function |
|---|---|
| ZXR10#**show iptv channel** {**all** \| **name** <*channel-name*>\| **idlist** <*channel-idliset*>} | This shows the channel information of IPTV. |
| ZXR10#**show iptv package** [{**package-name** <*package-name*>\| **package-id** <*package-id*>}] | This shows the information of iptv package. |
| ZXR10#**show iptv view-profile** [<*viewfile-name*>] | This shows the information of view profile. |
| ZXR10#**show iptv rule port** <*port-name*>[{**vlan-id** <*vlan-id*>\| **vlan-name** <*vlan-name*>}][**channel**][**package**] | This shows CRC rules. |
| ZXR10#**show iptv rule statistics** [ **rule-id** <*rule-id*>] | This shows CRC rule statistics. |
| ZXR10#**show iptv client** [{ **((port** < *port*> **)** \| **((NPC** < *slot-no*> **)**}][{ **((vlan-id** < *vlan-id*> **)** \| **(( vlan-name** < *vlan-name*> **)**}] | This shows online IPTV users. |
| ZXR10#**show iptv channel statistics** [**channel-id** <*channel-id*>] | This shows channel statistics. |

This page is intentionally blank.

Chapter 15

# VBAS Configuration

**Table of Contents**

VBAS Overview ..... 149
Configuring VBAS ..... 149
VBAS Configuration Example ..... 150
VBAS Maintenance and Diagnosis ..... 150

## VBAS Overview

VBAS (VBAS) protocol is an extended inquiry protocol between IP-DSLAM and BRAS equipment. BRAS and IP-DSLAM use point-to-point link to communicate. Port information inquiry and response message are encapsulated in layer-2 Ethernet data frame.

Configure corresponding Digital Subscriber Line Access Multiplexer (DSLAM) of VLAN on BAS; in the course of PPPoE calling, start VBAS protocol, that is, mapping to corresponding DSLAM according to the VLAN in user band; BAS start user line identifier inquiry to DSLAM; DSLAM give user line identifier response to BAS. In this manual, the switches are DSLAMs.

VBAS function is implemented by sending VBAS messages between BAS and DSLAM.

## Configuring VBAS

To configure VBAS, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config)#`**vbas enable** | This enables VBAS globally |
| 2 | `ZXR10(config-vlan)#`**vbas enable** | This enables VBAS function in a designated VLAN |
| 3 | `ZXR10(config-if)#`**vbas trust** | This configures a VBAS |
| 4 | `ZXR10(config-if)#`**vbas port-type** {**user**\|**net**} | This configures a designated port as VBAS user port or network port |

**Note:**

- To disable VBAS, use **no vbas enable** command in global configuration mode.
- To disable VBAS in a designated VLAN, use **no vbas enable** command in vlan configuration mode.
- To close a trust port, use **no vbas trust** command in interface configuration mode.

# VBAS Configuration Example

This example describes how to start VBAS function on Switches. Configure VBAS and enable vlan as vlan1; configure fei_1/1 as trust port, its type is user.

```
ZXR10(config)#vbas enable
ZXR10(config)#vlan 1
ZXR10(config-vlan)#vbas enable
ZXR10(config-vlan)#exit
ZXR10(config)#interface fei_1/1
ZXR10(config-if)#vbas trust
ZXR10(config-if)#vbas port-type user
```

# VBAS Maintenance and Diagnosis

To configure of maintenance and diagnosis, use the following command.

| Command | Function |
|---|---|
| ZXR10#**debug vbas** | This starts VBAS debug function and outputs the debug information |

Chapter 16

# CPU Attack Protection Configuration

**Table of Contents**

CPU Attack Protection Overview..... 151
CPU Attack Protection Principle..... 152
Configuring CPU Attack Protection..... 152
CPU Attack Protection Configuration Examples..... 154

## CPU Attack Protection Overview

Wide use of Internet and IP technology are bringing great changes to the world. With great benefits from IP network for life and work, there is also great loss due to attacks in network and computer virus invading. In the past, network attack and virus aim at PCs and servers. But now, network attack and virus also begin to aim at network devices, such as switches and routers.

For switch, it is possible to take protection measure according to known or predictable network attack and virus. This makes the switch have ability to protect itself and guarantee network security.

CPU attack protection function is to monitor upward rate of packets. When discovering packets with abnormal upward rate, system makes alarm. This prompts network management that there may be packets attacking CPU. Network management system decides whether to discard this kind of packet or not according to situations. Or network management system filters unreasonable packets.

**CPU Attack Protection Working Principle**

If IPv4 or IPv6 protocol protection function is disabled, some kind of protocol packets are discarded by bottom layer drives directly. And some kind of protocol packets are transmitted to upward by bottom layer drives with lower priorities. When these packets reach MUX module, they are discarded, except SNMP packets and RADIUS packets. So platform is not shocked.

If IPv4 or IPv6 protocol protection function is enabled, protocol packets are transmitted to platform with high priorities. When protocol protection module discovers that some kind of protocol packets are transmitted to platform in a high rate, the module makes alarm. This warns users that there may be some kind of

protocol packets attacking CPU. When such alarm appears, disable protocol protection function to protect CPU from being attacked.

**Note:**

After protocol protection functions of SNMP and RADIUS are disabled, they are not affected and work normally.

For IPv4 and IPv6 protocols, there is a threshold value. By default, the threshold value is 3000, that is, system allows receiving 3000 messages of a protocol within 30 seconds. When there are more than 3000 messages received, alarm appears. The threshold value can be configured.

# CPU Attack Protection Principle

Protocol protection is to protect the CPU of a switch. If CPU is attacked by many protocol messages, CPU usage ratio will increase. When protocol messages are sent to CPU at a high speed, protocol protection module will count the protocol messages of each type. Controlled by a timer, the number of protocol messages sent to CPU during a cycle is compared with a configured threshold value. For example, the number of protocol messages sent to CPU within 30 seconds is bigger than the configured threshold value, system sends a piece of alarm information in format of "Receive too many packets of 'protocol message type' from port 'port number'". This indicates the user that there may be attack of some type of protocol message on a port. If the user considers this is an attack, the user can disable this type of protocol protection. Therefore, this type of protocol messages can not be sent to switch platform and can not attack CPU anu more. When the user considers that the attack stops, the user can enable protocol protection again and normal messages of this protocol can be sent to CPU to be processed.

# Configuring CPU Attack Protection

## Configuring IPv4 Protocol Protection

IPv4 and IPv6 protocol protection is configured in interface configuration mode. So it modifies this function of physical interfaces.

To configure IPv4 protocol protection, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config-if)#**ipv4 protocol-protect mode** <*protocolname*>{**enable**\|**disable**} | This sets IPv4 protocol protection function |
| 2 | ZXR10(config-if)#**ipv4 protocol-protect alarm mode** <*protocol name*>< *alarm-limit* > | This configures alarm limit of IPv4 protocol protection |
| 3 | ZXR10(config-if)#**ipv4 protocol-protect average-rate mode** <*protocol-name*><*10-600*> | This configures the average rate of IPv4 protocols |
| 4 | ZXR10(config-if)#**ipv4 protocol-protect peak-rate mode** <*protocol-name*><*100-1000*> | This configures the peak rate of IPv4 protocols |

**Note:**

IPv4 protocols that are supported by CPU attack protection include ospf, pim, igmp, vrrp, icmp, arpreply, arprequest, group mng, vbase, vrrp arp, dhcp, rip, bgp, telnet, ldp_tcp, ldp_udp, ttl=1, bpdu, snmp, msdp and radius.

# Configuring IPv6 Protocol Protection

To configure IPv6 protocol protection, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config-if)#**ipv6 protocol-protect mode** <*protocolname*>{**enable** \| **disable**} | This sets IPv6 protocol protection function |
| 2 | ZXR10(config-if)#**ipv6 protocol-protect alarm mode** <*protocol name*><*alarm-limit*> | This configures alarm limit of IPv6 protocol protection |
| 3 | ZXR10(config-if)#**ipv6 protocol-protect average-rate mode** <*protocol-name*><*10-600*> | This configures the average rate of IPv6 protocols |
| 4 | ZXR10(config-if)#**ipv6 protocol-protect peak-rate mode** <*protocol-name*><*100-1000*> | This configures the peak rate of IPv6 protocols |

**Note:**

IPv6 protocols that are supported by CPU attack protection include mld, na, ns, ra, rs, common icmp6, bgp6, rip6, ospf6, ldptcp6, ldpudp6, telnet6 and pim6.

# Configuring Layer 2 Protocol Protection

To configure Layer 2 protocol protection, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10(config-if)#**l2 protocol-protect mode** <*protocolname*>{**enable** \| **disable**} | This sets Layer 2 protocol protection function |
| 2 | ZXR10(config-if)#**l2 protocol-protect alarm mode** <*protocolname*><*alarm-limit*> | This configures alarm limit of Layer 2 protocol protection |
| 3 | ZXR10(config-if)#**l2 protocol-protect average-rate mode** <*protocol-name*><*10-600*> | This configures the average rate of Layer 2 protocols |
| 4 | ZXR10(config-if)#**l2 protocol-protect peak-rate mode** <*protocol-name*><*100-1000*> | This configures the peak rate of Layer 2 protocols |

**Note:**

Layer 2 protocol supported by CPU attack protection is LLDP.

# CPU Attack Protection Configuration Examples

**Example** This example shows how to enable OSPF protection function and to set alarm limit to be 2500.

```
ZXR10#config terminal
ZXR10(config)#inter gei_1/1
ZXR10(config-if)#ipv4 protocol-protect mode ospf enable
ZXR10(config-if)#ipv4 protocol-protect alarm mode ospf 2500
```

**Example** This example shows how to enable ICMP6 protection function and to set alarm limit to be 3200.

```
ZXR10#config terminal
ZXR10(config)#inter gei_1/1
```

```
ZXR10(config-if)#ipv6 protocol-protect mode icmp enable
ZXR10(config-if)#ipv6 protocol-protect alarm mode icmp 3200
```

This page is intentionally blank.

Chapter 17

# URPF Configuration

**Table of Contents**

URPF Overview ..... 157
Configuring URPF ..... 158
URPF Configuration Example ..... 159
URPF Maintenance and Diagnosis ..... 160

## URPF Overview

URPF serves to prevent attacks with source address spoofing to the network. Term "Reverse" is relative to normal route search. A router will get destination address of the packet and search for a route to the destination once it receives a packet. It will forward the packet if such a route is found or simply discard the packet if there is no available route to the destination.

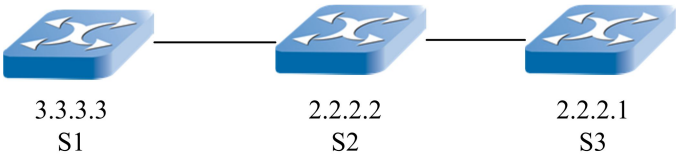
**Working Principle**

URPF gets the source address and ingress interface of the packet and uses source address as destination address to look up in the forwarding table and see if the interface corresponding to the source address matches the ingress interface. When interface does not match the ingress interface, it will regard source address as a false address and then discard the packet. In this way, URPF can effectively prevent malicious attacks by modifying the source address to the network.

**Module 1**

A simple network module is shown in Figure 37.

**FIGURE 37 SOURCE ADDRESS SNOOPING 1**



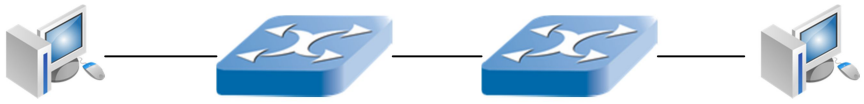

When S1 uses a packet with a false source address 2.2.2.1 to initiate a request to Server S2 which will send the packet to real address 2.2.2.1 (that is, S3) while responding to the request. This illegal packet will attack both S2 and S3.

Attackers may wage an attack by randomly changing source address in the packet. In this example, source address is one of reserved non-global IP addresses and thus is unreachable. A legal

IP address may also be used to wage an attack as long as it is unreachable.

**Module 2** Another network model is shown in Figure 38.

**FIGURE 38 SOURCE ADDRESS SNOOPING 2**



204.69.207.0/24

The attacker may forge a source address that is the address of another legal network and exists in global routing table. For example, attacker may forge a source address so that the attacked will think that the attack comes from forged source address but in fact source address is completely innocent. In addition, sometimes network administrator will close all data flows coming from that source address and this in return makes DOS attack of the attacker successfully become true.

A more complex scenario is that TCP SYN flooding attack will cause TCP SYN-ACK data packet to be sent to many hosts completely independent of the attack and such hosts will become victims. As a result, attacker may spoof one or more systems at the same time.

Similarly, UDP and ICMP may be used to implement flooding attacks.

All these attacks will severely lower the system performance or even cause system to crash. URPF is a technology to guard against such attacks.

# Configuring URPF

There are three types of URPF: Strict URPF (SRPF), Loose URPF (lRPF) and URPF that ignores the default route (lnRPF).

To configure URPF, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | `ZXR10(config-if)#`**ip verify** {**strict** \| **loose** \| **loose-ingoring-default-route**} | This enables the URPF check function on an interface |
| 2 | `ZXR10(config-if)#`**urpf log** {**on** \| **off**} | This enables or disables the URPF log function |

**Note:**

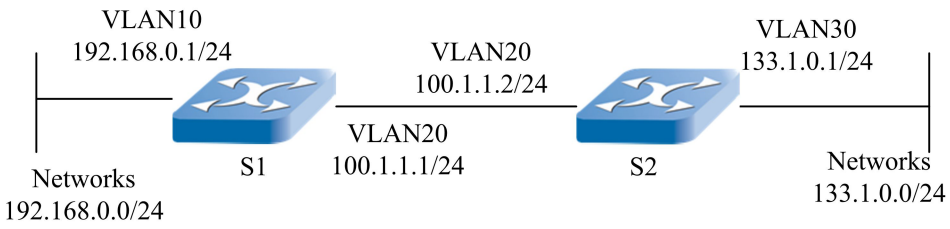
In step 1, the parameters are described below.

- **Strict** means that if egress port found by source IP address is different from data ingress port, it will be discarded; otherwise it will be processed in primary way.
- **Loose** means that if source IP address can find route, and egress port and ingress port of default route are coincident, it will be processed in the normal way, otherwise it will be discarded.
- **Loose-ingoring-default-route** means that if source IP address can find route and the route is not by default, it will be processed in the normal way. Otherwise it will be discarded.

# URPF Configuration Example

URPF network topology is shown in Figure 39.

**FIGURE 39 URPF CONFIGURATION EXAMPLE**




Strict URPF is configured on interface fei_1/2 on S1 so as to prevent the users behind network 192.168.0.0/24 from maliciously attacking networks behind S1.

Configuration on S1:

```
ZXR10(config)#interface fei_1/2
ZXR10(config-if)#sw ac vlan 10
ZXR10(config-if)#ip verify strict
ZXR10(config-if)#exit
ZXR10(config)#int vlan 10
ZXR10(config-if)#ip address 192.168.0.1 255.255.255.0
```

# URPF Maintenance and Diagnosis

To configure maintenance and diagnosis of URPF, perform the following steps.

| Step | Command | Function |
|---|---|---|
| 1 | ZXR10#**show interface** | This shows statistical count of URPF on an interface |
| 2 | ZXR10#**show ip traffic** | This shows the statistical count of URPF in the system |

Chapter 18

# IPFIX Configuration

**Table of Contents**

IPFIX Overview ..... 161
Configuring IPFIX ..... 163
IPFIX Configuration Example ..... 166
IPFIX Maintenance and Diagnosis ..... 166

# IPFIX Overview

## IPFIX Overview

IPFIX (IP Flow Information Export) is used to analyze and perform statistics to communication traffic and flow direction in network. In 2003, IETF select Netflow V9 as IPFIX standard from 5 candidate schemes.

To analyze and perform statistics to data flow in network, it is needed to distinguish types of packets transmitted in network.

Due to non-connection oriented characteristics of IP network, the communication of different types of services in network can be a series of IP packets sent from one terminal device to another terminal device. This series of packets actually forms one data flow of a service in carrier network. If management system can distinguish all flows in the entire network and correctly record transmit time of each flow, occupied network port, transmit source/destination address and size of data flows, traffic and flow direction of all communications in the entire carrier network can be analyzed and performed with statistics.

By telling differences among different flows in network, it is available to judge if two IP packets belong to the same one flow. This can be realized by analyzing 7 attributes of IP packet: source IP address, destination IP address, source port id, destination id, L3 protocol type, TOS byte (DSCP), ifIndex for network device input (or output).

With above 7 attributes of IP packet, flows of different service types transmitted in network can be rapidly distinguished. Each distinguished data flow can be traced separately and counted accurately, its flow direction characteristics such as transmit direction and destination can be recorded, and the start time, end time, ser-

vice type, contained packet number, byte number and other traffic information can be performed statistics.

As a macro analysis tool for network communication, Netflow technology doesn't analyze the specific data contained in each packet in network, instead it tests characteristics of transmitted data flow, which enables Netflow technology with good scalability: supporting high-speed network port and large-scale telecom network.

As for processing mechanism, IPFIX introduces multi-level processing procedures:

- In preprocessing stage, IPFIX can filter data flow of a specific level or perform sampling to packets on high-speed network interface based on demands of network management. With IPFIX, processing load of network device can be relieved and scalability of system can be enhanced while the needed management information is collected and performed statistics.
- In postprocessing stage, IPFIX can select to output all collected original statistics of data flow to upper-layer server for data sorting and summary; alternatively, network device can perform data aggregation to original statistics in various modes and send the summary statistics result to upper layer management server. The latter one can reduce the data quantity output by network device, thus decreasing requirement to configuration of upper layer management server and promoting scalability and working efficiency of upper layer management system.

IPFIX outputs data in format of template. Network device will send packet template and data flow records respectively to upper layer management server when outputting data in IPFIX format. Packet template specifies format and length of packet in subsequently sent data flow record for management server processing subsequent packets. Meanwhile to avoid packet loss and errors in packet transmission, network device repeats sending packet template to upper layer management server regularly.

# Sampling

IPFIX supports packet number-based sampling as well as time-based sampling. Sampling rate can be configured on each interface separately.

# Timeout Management

As for collected flow data,

- In case data are not updated within the inactive time, data will be output to NM server;
- As for long time active flow, the data will also be output to NM server after active time.

## Data Output

After collecting data flows in network, network device always outputs them to NM server. IPFIX supports to output data to multiple NM servers. Generally, data are output to two servers: master server and slave server.

IPFIX adopts template-based data output mode. IFPIX supports to send template every a few packets or at a certain interval. Packet template specifies the format and length of packets in subsequent data flows, and server resolves subsequent data flows according to template.

# Configuring IPFIX

## Basic Configuration

### Enabling/Disabling IPFIX Module

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream** {**enable**\|**disable**} | This enables/disables IPFIX module. |

### Setting IPFIX Memory Entries

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream cache entries** <*number*> | This sets the number of data flow entries stored in IPFIX module, 4096 by default. |

### Setting Aging Time of Active Stream

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream cache actinve** <*number*> | This sets aging time of active stream. |

As for long time active stream, in case it exceeds the set aging time, this data flow will age out, in minutes, 30 minutes by default.

## Setting Aging Time of Inactive Stream

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream cache inactive** <*number*> | This sets aging time of inactive stream. |

If data of a flow are not updated within the specified time, the aging information will be notified to stream record, in seconds, 15 seconds by default.

## Setting Sampling Rate

| Step | Command | Functions |
|---|---|---|
| 1 | ZXR10(config)#**interface** < *interface-name*> | This enters interface configuration mode. |
| 2 | ZXR10(config-if)#**netflow-sample** {**ingress**\|**egress** } | This configures packet number-based IPFIX sampling rate. |

## Setting NM Server Address and L4 Port ID

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream export destination** <*ip-address*> **udp-port** | This sets the address and port id of NM server, to which packets are sent. |

## Setting Source Address for Network Device Sending Packets

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream export source** <*ip-address*> | This sets source address for network device sending packets. |

## Setting Template Refresh Rate

| Step | Command | Functions |
|---|---|---|
| 1 | ZXR10(config)#**ip stream template refreh-rate number** | This sets the number of packets, after which template packet is sent, 20 by default. |
| 2 | ZXR10(config)#**ip stream template refreh-rate number timeout-rate number** | This sets template refresh rate time, 30 minutes by default. |

### Configuring TOPN

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream topn N sort-by** {**bytes**\|**packets**} | This sets size and sorting behavior of TOPN (by packet number or byte number). |

# Template Configuration

### Setting Template

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream templat template-name** | This sets template. |

### Setting Data Field Contained in Template Packet

| Command | Functions |
|---|---|
| ZXR10(config)#**match field** | This sets data field contained in template packet. |

Server resolves data contained in subsequent data flow according to these fields. The fields include source IP, destination IP, source port, destination port, the number of bytes contained in data flow, the number of packets contained in data flow, type of L3 protocol, TOS field, start time of data flow, end time of data flow, data flow ingress index, data flow egress index and TCP flag.

### Deleting Template

| Command | Functions |
|---|---|
| ZXR10(config)#**no ip stream template template-name** | This deletes one template. |

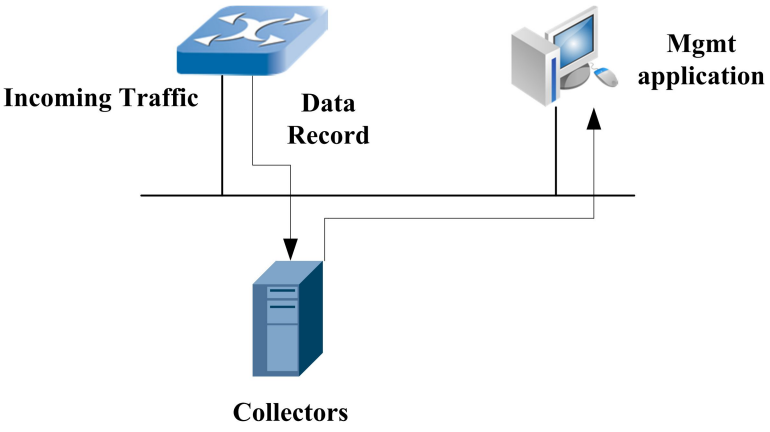
### Running Template

| Command | Functions |
|---|---|
| ZXR10(config)#**ip stream template template-name** | This runs template. |

# IPFIX Configuration Example

An IPFIX configuration example is given here with network topology as shown in Figure 40.

**FIGURE 40 IPFIX CONFIGURATION EXAMPLE**




```
ZXR10_R1(config)#ip stream enable
ZXR10_R1(config)#interface gei_2/12
ZXR10_R1(config-if)#netflow-sample ingress unicast 100
ZXR10_R1(config-if)#netflow-sample egress unicast 100
ZXR10_R1(config)#ip strem exprot destination 192.168.1.1 2055
ZXR10_R1(config)#ip strem exprot destination 192.168.1.2 2055
ZXR10_R1(config)#ip stream export source 192.168.1.244
ZXR10_R1(config)#ip stream export version 9
ZXR10_R1(config)#ip stream topn 10 sort-by packets
ZXR10_R1(config)#ip stream template test
ZXR10_R1(config-stream-tempalte)#match srcaddr
ZXR10_R1(config-stream-tempalte)#match dstaddr
ZXR10_R1(config-stream-tempalte)#match srcport
ZXR10_R1(config-stream-tempalte)#match dstsrcport
ZXR10_R1(config-stream-tempalte)#exit
ZXR10_R1(config)#ip stream run template test
```

# IPFIX Maintenance and Diagnosis

For the convenience of IPFIX maintenance and diagnosis, IPFIX provides related view commands.

1. To show IPFIX-related configurations, execute the following command:

   **show ip stream-config**

   This includes whether to enable IPFIX module, size of memory entries, server address, port configuration, source address configuration, template refresh rate and refresh time configuration.

2. To show TOPN, execute the following command:

   **show ip stream-topn**

   This shows information of N data flows according to set TOPN display mode. The information includes data flow ingress, egress, source address, destination address, source port, destination port, L3 protocol type, the number of packets or the number of bytes (corresponding to TOPNS setting).

3. To show template configuration, execute the following command:

   **show ipstream-template**

   This shows configuration of template, that is, fields contained in template.

This page is intentionally blank.

# Figures

Figure 1 Configuration Modes ..... 3
Figure 2 HyperTerminal Configuration 1 ..... 4
Figure 3 HyperTerminal Configuration 2 ..... 5
Figure 4 HyperTerminal Configuration 3 ..... 5
Figure 5 Running Telnet..... 7
Figure 6 Telnet Login Schematic Diagram..... 7
Figure 7 Telnet Connection Limit Configuration Example..... 9
Figure 8 Setting IP Address and Port of SSH Server.....10
Figure 9 Setting SSH Version .....11
Figure 10 WFTPD Window .....20
Figure 11 User/Rights Security Dialog Box .....21
Figure 12 TFTPD Window .....22
Figure 13 Configuration Dialog Box .....22
Figure 14 CLI Privilege Classification Function.....38
Figure 15 Port Mirroring Configuration Example .....53
Figure 16 ERSPAN Example.....54
Figure 17 ERSPAN Configuration Example .....55
Figure 18 Port Loop Detection Configuration Example .....58
Figure 19 DHCP Server Configuration Example .....68
Figure 20 DHCP Relay Configuration Example .....69
Figure 21 DHCP Snooping Preventing False DHCP Server.....70
Figure 22 DHCP Snooping Preventing Static IP.....71
Figure 23 Basic VRRP Configuration Example.....75
Figure 24 Symmetric VRRP Configuration Example .....76
Figure 25 Configuring Event Linkage ACL Rule .....86
Figure 26 ACL Configuration Example .....88
Figure 27 Traffic Monitoring Working Flow .....92
Figure 28 Typical QoS Configuration Example ..... 110
Figure 29 Policy Routing Configuration Example ..... 111
Figure 30 Dot1x Radius Authentication Application ..... 117
Figure 31 Dot1x Relay Authentication Application..... 118
Figure 32 Cluster Management Network ..... 122
Figure 33 Switching Rule ..... 123
Figure 34 Cluster Management Configuration Example..... 126

Figure 35 NTP Configuration Example ..... 130
Figure 36 LLDP Configuration Example ..... 139
Figure 37 Source Address Snooping 1 ..... 157
Figure 38 Source Address Snooping 2 ..... 158
Figure 39 URPF Configuration Example ..... 159
Figure 40 IPFIX Configuration Example ..... 166

# Tables

Table 1 CHAPTER SUMMARY ..... i
Table 3 Parameter Values..... 6
Table 4 Command Modes .....12
Table 5 IP Address for Each Class .....59
Table 6 ACL Descriptions .....78

This page is intentionally blank.

# List of Glossary

**AAA - Authentication, Authorization, and Accounting**
**ACL - Access Control List**
**ARP - Address ResolutionProtocol**
**BAS - Broadband Access Server**
**BOOTP - BOOTstrap Protocol**
**CBS - Committed Burst Size**
**CIR - Committed Information Rate**
**CLI - Command Line Interface**
**CoS - Class of Service**
**DHCP - Dynamic Host Configuration Protocol**
**DSCP - Differentiated Services Code Point**
**DSLAM - Digital Subscriber Line Access Multiplexer**
**DWRR - Deficit Weighted Round Robin**
**EAPOL - Extensible Authentication Protocol Over LAN**
**EBS - Excess Burst Size**
**FTP - File Transfer Protocol**
**ICMP - Internet Control Message Protocol**
**IP - Internet Protocol**
**IPTV - Internet Protocol Television**
**LLDP - Link Layer Discovery Protocol**
**LLDPDU - Link Layer Discovery Protocol Data Unit**
**MAC - Media Access Control**
**MIB - Management Information Base**
**NMS - Network Management System**
**NTP - Network Time Protocol**
**PBS - Peak Burst Size**
**PIR - Peak Information Rate**
**PVID - Port VLAN ID**
**QoS - Quality of Service**
**RADIUS - Remote Authentication Dial In User Service**
**RARP - Reverse Address Resolution Protocol**
**RFC - Request For Comments**
**RMON - Remote Monitoring**
**SNMP - Simple Network Management Protocol**
**SP - Strict Priority**

**SSH - Secure Shell**
**TCP - Transmission Control Protocol**
**TELNET - Telecommunication Network Protocol**
**TFTP - Trivial File Transfer Protocol**
**TLV - Type Length Value**
**ToS - Type Of Service**
**UDLD - UniDirectional Link Detection**
**UDP - User Datagram Protocol**
**URPF - Unicast Reverse Path Forwarding**
**VBAS - Virtual Broadband Access Server**
**VLAN - Virtual Local Area Network**
**VRRP - Virtual Router Redundancy Protocol**
**WRR - Weighted Round Robin**